The Impact of Imposing a Cost on Morphological Complexity for Teams of Cooperating Robots

Danielle Rose Nagar
University of Cape Town
Cape Town, South Africa
ngrdan001@myuct.ac.za

ABSTRACT

This is a study on the benefits of imposing a cost on morphological (sensory configuration) complexity while evolving controller-morphology couplings for co-operative robot teams. Namely, we investigate whether indirect-encoded evolutionary mechanisms discern simpler morphology for competent team behaviour when a cost is imposed on morphological complexity. Moreover, we are interested in the relationship between task (environmental) difficulty and the degree of morphological complexity. Towards this end, we report a novel method, *HyperNEAT-M*, for co-evolving controller-morphology couplings for robot teams - with additional support for multi-objective optimization as to allow for a cost of complexity to be imposed. Results indicate that constraining complexity induces simpler morphologies with negligible differences in task performance. Moreover, robot teams consistently favour simpler morphologies across tasks of varying difficulty.

1 INTRODUCTION

Towards a future in which robot design is fully automated, an open problem in cooperative Evolutionary Robotics (ER) [17] is how best to co-evolve robot controllers and morphologies to produce optimal sensory-motor configuration [11, 37, 53, 54, 62]. Co-evolutionary methods are particularly effective for the design of *Multi-Robot Systems* (MRS) [34]. In these systems robots interact via *emergent*, collaborative behavior. This is applicable to a number of real-world cooperative tasks [26, 36, 44]. Indeed, recent work attests to a direct correlation between improved task performance and increasing the extent to which robot design is left to evolutionary methods [9, 13, 27].

Common approaches to deriving *behaviour-morphology* couplings are *direct* [27, 52] and *indirect* [62] encoded *neuro-evolution* [64], falling within the more general class of stochastic optimisation methods that take inspiration from natural evolution. Recent literature shows that by evolving organisms *in silico* with an indirect encoded neuro-evolution method, artificial evolution naturally induces a selection pressure that favors solution complexity, beyond what is required to handle the environmental scenario [4]. Moreover, it was shown that imposing a *cost* on complexity (as is thought to occur in natural evolution [22, 43]) causes evolution to produce solutions which are only as complex as the environment requires them to be [3, 32, 42].

These findings could have important implications for robot design. Namely, constraining the complexity of evolved robot morphologies could provide solutions with cheaper, fewer and more desirably configured parts (such as sensors). This may result in reducing time and costs spent on the overall design process. Indeed, a number of studies corroborate the benefits of imposing an evolutionary cost on complexity for single-robot systems [3, 42]. This area of research could facilitate more cost-effective MRS design since redundant morphological complexity amplifies costs as the number of required robots increases. However, to the best of our knowledge, no such investigation has been conducted for MRS.

To facilitate this research, a chosen evolutionary method should allow for a cost of complexity to be incorporated into the optimization process. In lieu of standard single-objective neuro-evolution [23] techniques which are typically employed for ER research, multi-objective neuro-evolution techniques are designed to optimize multiple conflicting objectives simultaneously, taking into account the trade-offs among the objectives [18, 20]. This is especially suitable for robotic system design as many real-world problems contain various trade-offs and optimal trade-offs are contingent on the problem domain. An obstacle to this research is that prominent neuro-evolution methods, notably Neuro-evolution of Augmenting Topologies (NEAT) [59] and HyperNEAT [58], are designed for single-objective optimization. Several attempts to extend these methods for multi-objective optimisation are reported in the literature, but for the most part are without quantitative support for nor preservation of the core innovations of the original methods [2]. One such exception is *NEAT-MODS* [2] which is intended to be used in this work to facilitate the imposition of a cost on complexity.

This work is an inquiry into the potential benefits of imposing a cost on morphological complexity in the indirect co-adaptation of controller and morphology for homogeneous MRS. Namely, we are interested in (1) producing cheaper sensory (morphological) configurations without reducing task performance and (2), investigating what degree of sensory complexity is necessary for environments of varying difficulty. Towards this end, we establish a novel method *HyperNEAT-M* that extends HyperNEAT to evolve the controller in addition to the sensory-configuration. This method is compared to *HyperNEAT-M-MODS* which integrates HyperNEAT-M with *NEAT-MODS* to allow for additional selection pressure towards morphological simplicity. We utilize *collective gathering* [39] - a benchmark task in cooperative robotics [38] - to quantify task performance. The potential benefits of constraining morphological complexity are not limited to a single problem domain, and so we focus on the capacity to perform the benchmark task. We define *morphological complexity* as an abstraction of the design costs associated with additional and more powerful sensors. This work evolves behaviorally and morphologically homogeneous MRS due to the increased computational complexity required for heterogeneous robots.

We investigate two hypotheses :

(0) *Imposing a cost on morphological complexity on indirect evolutionary processes results in selection for simpler morphologies with negligible adjustments of task performance for MRS.* This hypothesis is based on previous research which concurs that indirect evolutionary processes tend to induce selection pressure for increased complexity [4]. Moreover, related research demonstrated that increasing morphological complexity does not result in higher task performance [61, 62].

(1) *Increasing environmental complexity does not require an increase in morphological complexity for Multi-Robot Systems* Similar works have demonstrated that for single-robot systems, adding a cost on complexity results in a relationship between environmental complexity and morphological complexity [4, 5, 42]. However, this hypothesis is based on various studies, within the field of MRS, that have implicated that increasing task complexity does not necessarily correlate to increasing morphological complexity [27, 61, 62].

2 BACKGROUND

2.1 Neuro-evolution

Neuro-evolution (NE) [23] is a sub-field of machine learning that unifies two biologically-inspired techniques: Evolutionary Algorithms (EAs)[20] and Artificial Neural Networks (ANNs)[33]. NE exploits the meta heuristic approach of EAs, to generate and evolve ANNs[50, 64], culminating in one of the most effective methodologies for producing complex and novel controllers [25, 51, 65]. Two particularly powerful NE algorithms are Neuro-evolution of Augmenting Topologies (NEAT) algorithm [59] and HyperNEAT [58] which not only evolve the synapses of the ANN but also its topology.

2.1.1 Neuro-evolution of Augmenting Topologies. NEAT [59] is a direct encoded genetic algorithm that is based on three fundamental principles: (1) Historical markings allowing for appropriate recombination, (2) speciation to protect innovative solutions, and (3) complexification to favor minimalistic solutions.

NEAT adapts a genetic encoding schema that represents ANNs as a list of *connection genes.* A connection gene stores pertinent information including an *innovation number* - a historical marker that represents the gene's origin. These markers allow for sensible recombination, as genes can be matched and appropriately chosen from the respective parents. However, the addition of novel genes to a solution causes an initial decrease in overall fitness. These solutions require additional time to optimize before being able to fairly compete with other solutions. NEAT solves this by ensuring competition for selection is held within groups of similar organisms (*species*) rather than the entire population. Speciation is used to categorize solutions based on their topology and synapses which can be achieved trivially via historical markings. The final principle element of NEAT is complexification, where the initial population of solutions are uniform and contain no hidden layers. This ensures that all additional complexity is justified and gives NEAT a performance advantage over related approaches, such as SANE [25, 35].

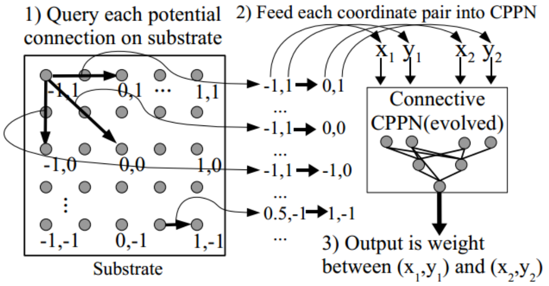

Figure 1: Querying a CPPN with a substrate to produce a connectivity pattern in space. Each potential connection (node pair) in the substrate topology on the left is passed to the CPPN, which calculates a weight for that pair. If the weight computed for a potential connection satisfies a certain threshold, then that connection will be 'painted' in n-dimensional space (two dimensional in this instance) [58].

NEAT has shown to be a particularly powerful methodology for multiple ER and MRS applications such as evolving autonomous cars [63], simulating video games [52] and others [28].

2.1.2 HyperNEAT. HyperNEAT [58] is an extension of NEAT. Its key innovation is its ability to exploit the physical underlying structure inherit to the problem [57]. HyperNEAT utilizes a generative indirect encoding schema where the genotype representation of an ANN is a connective Compositional Pattern Producing Network (CPPN) [57]. CPPNs represent connectivity patterns as functions of Cartesian space. They are graph structures internally similar to ANNs but differ in their utilization of multiple activation functions. The composition of these functions allows complex, repetitive and regular patterns to be produced. Moreover, the structure allows neuro-evolution methods specifically NEAT to evolve CPPNs.

A CPPN is decoded into an ANN through the painting of synapses' connections on an *a priori* defined substrate. A CPPN achieves this by accepting the Cartesian co-ordinates of two nodes from the substrate and outputting a weight for the respective connection. However, a connection is only solidified if the weight produced is above a certain threshold [58]. Thus, a two-dimensional connectivity pattern (ANN) can be represented by a CPPN's spatial pattern in a four-dimensional hypercube. This is presented in figure 1. This process is specifically advantageous as it allows for compact representation of large ANNs.

In this study, HyperNEAT is selected over NEAT for its increased learning capacities and ability to exploit the geometric regularity and modularity of a problem. In the chosen collective gathering task, the geometric features include the sensory configuration, resources (blocks) as well as the relative position and direction of other robots. The numerous advantages of HyperNEAT for multi-agent tasks are demonstrated in its applications to related areas of research, such as for *RoboCup Soccer* [16], *Pursuit-Evasion* [37] and collective construction [47, 61].

2.2 Multi-objective Evolutionary Algorithms

Within the research area of MRS, tasks may require multiple competing objectives to be simultaneously optimized [18], a question arises of how to consolidate them. Moreover, designers seek to understand the trade-offs between the various objectives and hand pick their preferred solution [60].

Multi-objective Evolutionary Algorithms (MOEA) [10] are an increasingly popular solution to this problem. State-of-the-art MOEAs produces a set of *Pareto-optimal solutions* that embody optimal trade-offs between several objectives [1, 20]. A candidate solution is Pareto-optimal if there exists no other solution that could perform better at a single objective, without decreasing the performance of other objectives [18]. EAs are a favorable method for solving multi-objective problems as they naturally process families of solutions and thus can generate, in one run, a set of solutions. Moreover, EAs are efficiently able to process discontinuous and concave pareto-fronts [14].

However, NEAT and HyperNEAT were developed for single optimisation problems. With an increased demand for MOEAs, integrating these has become an active focus of contemporary research [2]. It is important to note that such an implementation will alter the core NEAT structure which HyperNEAT is built upon. Thus, any multi-objective NEAT integration can trivially be extended to HyperNEAT. A core issue in this integration is maintaining the principle structure of NEAT while simultaneously preserving core features from state-of-the-art MOEs mainly NSGA-II [56] and SPEA-II[66]. The main contentious issue is the selection mechanism - NEAT requires selection be done through speciation while MOEAs require a form of non-dominance sorting (mainly NSGA-II). Furthermore, successful MOEAs maintain the principle of elitism, where parent solutions and child solutions compete for the next generation [56]. This ensures that pareto-optimal solutions are carried through to each generation. The literature points to various integration strategies:

2.2.1 Augmenting the Fitness Function. One prominent early strategy is the alteration of the fitness function to take into account a solution's pareto-optimality. This is demonstrated in the NEAT-PS algorithm [63], which adapts the Pareto-strength value approach from SPEA-II into the fitness function. NEAT-PS was successfully utilized to find the optimal trade-offs between comfort (fewer lane changes) and speed (more lane changes) in vehicle controllers [63]. A contention with such a technique is since NEAT does not follow an elitist strategy, NEAT-PS does not either. Consequently, Pareto-optimal solutions from previous generations are not stored. Another alternative worth mentioning, is a single fitness function that assigns appropriate weights to each objective [14]. However, this is not ideal for contradictory objectives as trade-offs are not taken into account.

2.2.2 Genotypic diversity function. More modern works have opted to remove NEAT's selection process, and consequently speciation, in order to incorporate core NSGA-II functionality [4, 52]. Subsequently, an additional genotypic diversity function is added in order to protect innovation, which is traditionally achieved via speciation. There are three fundamental concerns with this algorithm. (1) whether the removal of speciation damages the credibility of NEAT. (2) there may be issues that arise from a genotypic diversity function. In particular, solutions may advance through generations for their novelty as opposed to their ability to perform well at the task. (3) the increase in dimensionality of the objective space is not ideal for already complex problems [2]. Such an algorithm was implemented by a key work that heavily influenced this paper [4] and other alternative papers that require a multi-objective version of NEAT [30, 31, 52].

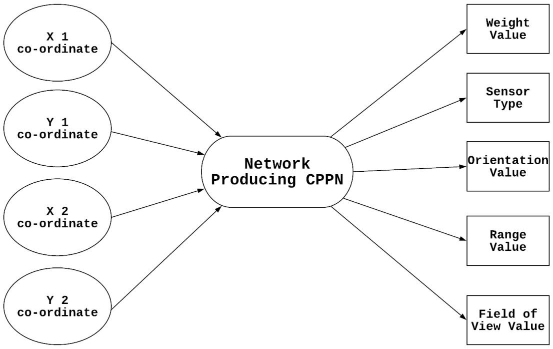

Figure 2: HyperNEAT-M CPPN: The CPPN maps the Cartesian coordinates of two nodes from the substrate to a connection weight along with a sensory configuration (a sensory type, orientation, range and FOV value).

2.2.3 NEAT-MODS. In contrast, a recent paper by Abramovich et al. [2] presents NEAT Multi-objective Diversified Species (NEAT-MODS), an elitist multi-objective NEAT structure that is able to consolidate both NSGA-II selection process while still maintaining all core principles of NEAT including speciation. Additionally, the paper demonstrated that NEAT-MODS was substantially more effective than NEAT-PS [2]. Although other papers have claimed to have also implemented a version of NEAT that also achieves this [12], none have presented the algorithmic structure with comparative testing to alternative methodologies. This paper [2], to the best of our knowledge, is the only well-formulated and generic multi-objective NEAT alternative. Consequently, this work adopts a version of this algorithm for multi-objective HyperNEAT. A more in depth discussion on the algorithm can be found in **??**.

3 METHODS

Two main methods are used to allow for the co-evolution of morphology and controller with one objective (task performance) and two objectives (task performance and morphological simplicity). The first method, *HyperNEAT-M* extends HyperNEAT to evolve the controller in addition to the sensory-configuration - specifically the number of sensors and their respective *Sensor Type, Field of View, Range, Orientation* and *Bearing*. The second method, *HyperNEAT-MODS*, is a multi-objective version of *HyperNEAT-M* allowing for an additional cost on complexity to be imposed.

3.1 Co-evolution of Robot Morphology and Controller

The related *NEAT-M* method [27] co-evolves controller and morphology by directly encoding a *morphology genotype* which is evolved in parallel to the usual controller genotype. HyperNEAT-M, however, indirectly encodes controller and morphology in a single CPPN genotype.

While a HyperNEAT CPPN computes only a connection weight value as a function of two Cartesian points (that is, two connected nodes), a HyperNEAT-M CPPN computes a *sensory configuration* constituted by five additional values: *range, bearing, field of view (FOV), orientation* and *type* as shown in figure 2. This is achieved by adding additional output nodes to the CPPN, similarly to another study [48] which encodes additional ANN parameters directly into the CPPN. The sensory configuration is computed by converting numerical CPPN output ([-1,1]) to a sensory parameter. More specifically, for categorical configurations (sensor type) we associate output ranges with a sensor type. Numerical sensory configurations (range, Bearing, Field of View (FOV), Orientation) are computed by normalizing the CPPN output against the range of the sensory parameter in accordance with the sensor type. For example, if the CPPN outputs 0.2 for a bearing value and the bearing range associated with the specific sensor type is [-2,2] then the bearing parameter will be computed as 0.4 (0.2 normalized to the range [-2,2]). Subsequently, each connection in the resulting network (phenotype) encoded by the CPPN stores a sensory configuration and the usual connection weight. Note that the bearing (that is, the location on the robot body) of a given sensor directly corresponds to the Cartesian co-ordinate of an input node on the substrate which receives environmental input from that sensor. This is illustrated in Figure 2, where *X 1 coordinate* and *Y 1 coordinate* constitute the Cartesian co-ordinate (on the physical robot body) of a given sensory input node.

The final configuration for each sensory input node is established by traversing the outgoing connections and setting the sensory parameters from connection with the highest weight. It is important to note that a sensor (input node) is only configured if there exists at least one outgoing connection.

3.2 Multi-Objective Co-evolution of Robot Morphology and controller

HyperNEAT-M-MODS supersedes the core foundation of HyperNEAT-M (NEAT) with *NEAT-MODS*, an NSGA-II based MOEA designed to adhere to the principles of NEAT while simultaneously allowing auxiliary objectives [2]. This is accomplished by augmenting the selection process of NEAT, to follow an elitist strategy of considering a combined population of the current offspring and respective parent solutions. This combined population is sorted via a non-domination sorting and crowding-distance as in NSGA-II. As well as placed into respective species. Subsequently, the selection process then occurs in two phases:

(1) **Selecting Species for the Next Generation:** A sorted list of selected species is produced by traversing the ordered solutions. Additionally, NEAT-MODS incorporates a bound on the number of selected species. This bound maintains global elitism while favoring diversity in the selection process.
(2) **Selecting Individuals from the Selected Species:** The final phase of the process aims to produce a new parent population from the selected species. Each selected species is re-sorted (based on NSGA-II's sorting mechanism) and a serial progression technique is applied. The serial progression technique, attempts to iteratively scan through each species and select a solution, until a final parent population has been selected. Thus, speciation is maintained as well as pareto-optimality.

A more thorough treatment of the algorithm can be found in the associated paper [2].

4 EXPERIMENTAL DESIGN

Two experiments are designed to investigate the implications of adding a cost to morphological complexity for co-evolved MRS. Firstly, experiment SO employs HyperNEAT-M with selection pressure based solely on task performance. Secondly, experiment MO adopts HyperNEAT-M-MODS in the handling of selection pressure for both task performance and morphological simplicity. Similar to related works [27, 47], we administer the experiments in a collective gathering simulator for teams of homogeneous robotic agents. Each experiment is run in three environments with varying degrees of collaboration (defined as task complexity) needed to complete the task. This section outlines the configurations used to conduct the above experiments.

4.1 Experimental Platform

The experimental framework used is a Mason based MRS simulator [1], developed for a collective gathering task with various configurations of task complexity. The robotic agents within the simulator are modeled after the Khepera III [29] robots and allow for varying sensor configurations. The simulator functions as a medium to determine the fitness of evolved morphology and controller solutions.

4.1.1 *Collective Gathering Task.*

Collective Gathering requires a team of robots to locate distributed resources in a bounded environment and transport them, via cooperative pushing, to a predefined zone [39]. The task was selected for its pertinence in real-world MRS applications such as toxic waste clean-up [44]. Moreover, collective gathering is a well established benchmark task in ER and acts as a performance indicator for various evolutionary design techniques [38, 39].

4.1.2 *Environmental Configuration.*

The experimental framework, displayed in figure 3, simulates a bounded two dimensional continuous environment with a target area for the robots to place collected resources (blocks). Three different types of blocks (small, medium and large) are randomly distributed within the environment. Larger blocks increase the degree of task complexity by requiring cooperative pushing. The transportation of small blocks require a single robot, medium blocks require two or more robots and large blocks require three or more robots. Correspondingly, each block type has a reward value (defined in table 2) and the fitness of a team of robots is defined by the total value of blocks that are in the target area at the end of the simulation.

The distribution of block types in an environment defines the task complexity. Within environments that contain mainly small blocks, robots are able to work concurrently and with minimum cooperation. Such environments have significantly lower task complexities than environments with more medium and large blocks where robots are required to work cooperatively to complete the task.

[1] https://cs.gmu.edu/~eclab/projects/mason/

Neuro-Evolution Parameters	
Generations per experiment	250
Trial evaluations per phenotype	5
Population size	150
Initial Connection Density	0.9
Substrate Input Nodes / Output Nodes	5 / 2
Simulation Parameters	
Time steps per simulated trial evaluation	10000
Repititions of each experiment	5
Robot team size	20
Robot size (diameter) / Gripping distance	0.004 / 0.002 (Portion of environment size)
Initial robot / block positions	Random (Outside gathering zone)
Environment width x height / Gathering zone size	1.0 x 1.0 / 0.5 x 0.2
Small / Medium / Large block size (Width / Height)	0.01 x 0.01 / 0.015 x 0.015 / 0.02 x 0.02
Ultrasonic sensor Range / FOV	$(0.0, 1.0]/(0.0, \pi)$
Infrared Proximity Range / FOV	$(0.0, 0.4]/(\frac{\pi}{6}, \frac{5\pi}{6})$
Colour Sensor Range / FOV	$(0.0, 0.4]/(\frac{\pi}{6}, \frac{5\pi}{6})$
Low Res Camera Range / FOV	$(0.0, 0.8]/(\frac{\pi}{9}, \frac{8\pi}{9})$
Sensor Orientation Range	$[\frac{\pi}{2}, \frac{\pi}{2}]$

Table 1: The neuro-evolution and simulator parameters

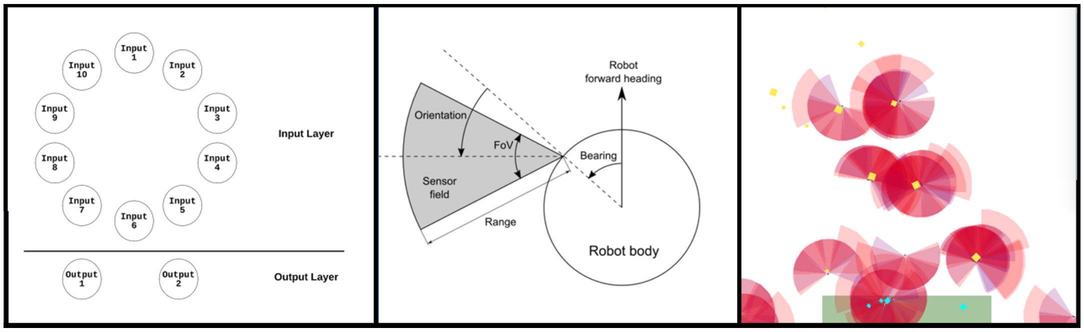

Figure 3: left: The configured HyperNEAT substrate that models the robotic agent geometry. middle: A top-down view of the simulated robotic agents with the sensory parameter configurations. right: An instance of the simulation environment for collective gathering. Robots search for randomly distributed type small, medium, and large blocks. The coloured semi circles represent the range and FOV of the robots.

4.1.3 Robot Configuration. The class of robotic agents utilized is a team of homogeneous robots. Specifically, all robots are controlled by the same ANN controller and sensory configuration, tasked with cooperating in order to complete the collective gathering task. The robots are modeled after the Khepera III robot as shown in figure 3. This model was chosen as they are prominent within the ER research field [24, 40, 41, 55] and allow for sensory modification.

The behavior of a robot is governed by the collaboration between its controller and sensor configuration. Each input node of a controller is associated with a sensor. The sensors feed appropriate data into the controller's input layer and outputs it to two wheel motors. The motor wheels control a robot's direction at a constant speed.

The robotic agent's sensory configuration is based on the framework defined in related works [27, 47]. Namely, a robot can utilize five types of sensors: an infrared proximity sensor, a colour proximity sensor, an ultrasonic sensor, a low resolution camera sensor and a bottom proximity sensor. Furthermore, as shown in figure 3, each sensor has four different parameters that can be specified namely Range, Bearing, Field of View, Orientation. In all experiments, the number of sensors, respective type and parameter configuration is put under the evolutionary control of HyperNEAT-M or HyperNEAT-M-MODS. It is also worth noting that the agents feature additional collective gathering heuristics. A thorough treatment of the simulator's heuristics, sensory and general robot parameters is provided by related works [27, 47], and our chosen parameters are listed in Table 1.

4.2 Experimental Configuration

4.2.1 Task Configuration. Three environments that demand varying degrees of task complexity, given in table 2, are defined. The environments require low, medium and high cooperation based on the distribution of block types (Environment 1, 2 and 3, respectively). These environmental configurations were chosen based on related research and facilitate the evolution of optimal emergent behavior [27]

	Blocks		
	Small (1)	Medium (2)	Large (3)
Environment 1	10	5	0
Environment 2	5	5	5
Environment 3	0	5	10

Table 2: Task environments have varying block configurations that require increasingly higher degrees of cooperation to complete the task. More specifically, environment 1 requires a lower amount of cooperation that environment 3. The brackets define a blocks value as well as the number of robots needed to move a given block.

4.2.2 Fitness Evaluation. In experiment SO, a team's fitness is evaluated solely on their collective gathering task performance. This takes into account the total reward value of blocks collected, where blocks that require higher cooperation yield progressively higher reward. A bonus reward is given to teams that collect all resources within the allocated simulation time (1000 time steps).

Thus, task performance was calculated at the end of each simulation as

$$ft = 100 \times \frac{v_c}{v_t} + 20 \times (1.0 - \frac{s_e}{s_t}) \qquad (1)$$

where v_c was the sum value of resources gathered, v_t the total value of all resources in the environment, s_c the total number of time-steps elapsed and s_t the total number of time-steps. A team's final fitness is the average task performance evaluated over five trial simulations. Note, similarly to related works [27], evolution is halted when a solution converges to maximizes task performance (collects all resources within half the allocated time steps). However, such a convergence mechanism is not used for experiment MO.

In experiment MO, ft is maximized in conjunction with minimizing morphological complexity through the use of HyperNEAT-MODS. The morphological complexity of a homogeneous robot team reflects the number of sensors n ($n \in [0, 10]$) as well as the FOV f_i and range r_i value of each sensor s_i. More specifically, we consider that depending on a sensors *type* (table 1)f_i and r_i are constrained. As such, we allocate each sensor a contribution to morphological complexity by examining the proportion selected to the maximum value that could be selected. Thus, morphological complexity is defined as

$$C = 100 - 5 \times \sum_{i=1}^{n} \left(\frac{f_i - \wedge F_i}{\vee F_i - \wedge F_i} + \frac{r_i - \wedge R_i}{\vee R_i - \wedge R_i} \right) \qquad (2)$$

where $\wedge F_i$ and $\vee F_i$ are the minimum and maximum FOV value for S_i's associated type and $\wedge R_i$ and $\vee R_i$ are the minimum and maximum range value for s_i's associated type. This effectively mimics real-world expenditure of a sensory configuration where sensors with a larger FOV and range are considerably more expensive.

4.2.3 Neuro-evolution Configuration. HyperNEAT is purposed with a circular substrate (figure 3) to mimic the physical positionality and symmetry of the sensory configuration. As mentioned above, the physical position of each input node maps to the bearing parameter of the associated sensor. All other respective neuro-evolution methods can be found in table 1.

4.3 Experiments

This study comprises of six experiments[2], where both experiments (MO and SO) are run in the three environmental configurations (table 3. For each experiment we co-evolve controller and morphology for a population of 150 robot teams over 250 generations. We evaluate each robot team over five trial simulations with each simulation testing random starting positions of the robots and blocks. Due to the high computational power and consequently the substantial amount of time (four days) required to run each experiment, we run each experiment five times.

	Environment		
	1	*2*	*3*
Experiment SO	S1	S2	S3
Experiment MO	M1	M2	M3

Table 3: The experiment ID's of the six experiments run in this study.

5 RESULTS AND DISCUSSION

In this section we present the results of our experiments. The results support the outlined hypothesis and conclude that the imposition of a cost of complexity offers numerous benefits for indirect co-adaptation of controller and morphology for MRS.

5.1 Hypothesis 0:

Imposing a cost on morphological complexity results in selection pressure for simpler morphologies with negligible adjustments of task performance.

Corroboration of hypothesis 0 requires comparison of solutions produced with and without a cost on complexity namely experiment SO and MO respectfully. Figure 4 validates the hypothesis and illustrates that in each environment there exists some solution on the Pareto front that offers a simpler morphology with negligible differences in task performance. This difference is due to the stochastic nature of evolutionary processes. We define this difference as 10%, the average standard deviation of performance across all experiments [4, 20]. However, there exists no established method for statistical comparison of a singular solution (produced by experiment SO) to a Pareto front (produced by experiment MO).

Traditionally, a single point is compared to an optimal Pareto front point selected via value judgment or by some selection process mechanism from the literature [8, 15, 21]. This is not an appropriate solution for this work as the selection of a point or mechanism is dependent on the problem domain. Instead, we opt to use a similar methodology to that of Auerbach and Bongard [4] and compare singular points to various Pareto optimal solutions elected by well-established selection process mechanisms. We utilize the following selection processes process mechanisms:

1. **Max ($\vee X$):** The solution with the best task performance [21].
2. **Mean ($\bar{X}$):** The average of all solutions (both objectives) on the Pareto front [4].

[2] The source code and experimental framework can be found at https://github.com/rudolfbono/honours-project.

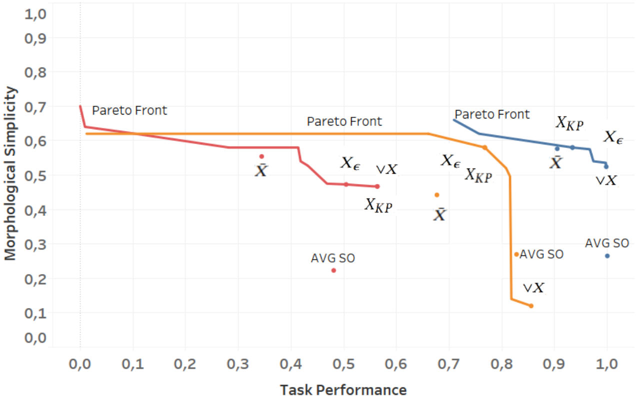

Figure 4: Compares the solutions (both objectives) evolved in experiment SO (AVG SO) to experiment MO (Pareto Front) over the three environments. Specifically, for experiment SO we average the candidate solutions produced over all runs and for experiment MO we perform a non-dominated sorting to all Pareto fronts producing an optimal Pareto front. Blue, orange and red represent environment 1, 2 and 3 respectfully. The definitions of the labels are outlined in section 5.2. Note: the simplest morphology ($M = 0$, equation 2) is associated with a full sensory configuration. The highest morphological simplicity ($M = 1$, equation 2) is associated with no sensory configuration.

(3) **Knee Point (X_{KP}):** The point with the smallest euclidean distance to the utopia point [15]. The utopia point is defined as the point where all objectives are maximized.

(4) **Epsilon (X_ϵ):** A point that maximizes sensory simplicity without decreasing task performance by more than a negligible difference (10%) from the $\vee x$ [21].

We compute these four points for each resultant Pareto front from each MO experiment. Two sets of student's t-tests (table 4) are conducted to compare the results of experiment SO to each of the four points for each environment. A student's t-test was chosen as the Shapiro-Wilk test showed that the data was normally distributed (p<0.05). The first test analyses significant differences in task performance and shows that, on average experiment SO outperformed experiment MO by no more than 10 percent, which is within the range of negligible difference. The second set of t-tests examined significant differences between morphological simplicity for MO and SO. Results showed that MO achieved greater morphological simplicity than SO in all cases (20 to 30 percent). These results are consistent with related works [61, 62] and imply that applying cheaper sensory configurations to MRS does not result in reduced task performance. This is theorized to be a result of the hypothesis that indirect evolutionary processes tend to favour a complexity above what is required to perform a task [4]. Thus, similarly to single-robot systems [3], imposing a cost of complexity can reduce robotic design costs for MRS without decreasing task performance. An interesting observation (table 4) is that as environmental difficulty increases experiment SO and MO seem to display increasingly similar hypothesized means. Why this occurs remains the topic of ongoing research.

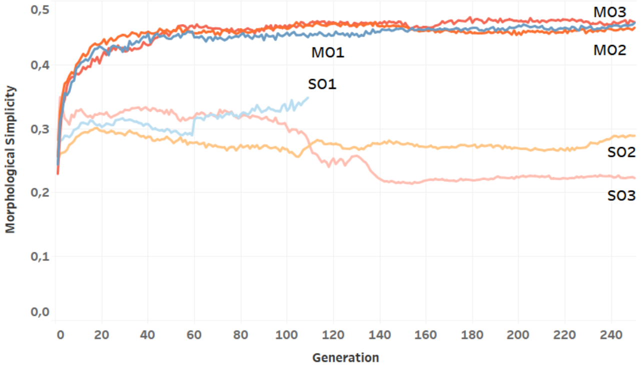

Figure 5: The mean morphological simplicity calculated across generations for experiment MO and SO. Blue, orange and red represent environment 1, 2 and 3 respectfully. Darker colours represent experiment MO and lighter colours represent experiment SO. The labels represent the experiment labels 3. Note: the experiment SO1 converged around the 110th generation

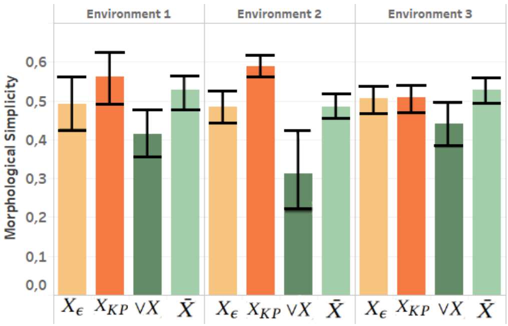

Figure 6: The mean morphological simplicity (and standard error) of each Pareto selection point across environments.

5.2 Hypothesis 1

Increasing environmental complexity does not require an increase in morphological complexity for Multi-Robot Systems

A corollary of hypothesis 0 is that imposing a cost on complexity produces an approximation of the simplest sensory configuration required for behavioural competence *in a given environment*. However, in order to understand how the relationship between morphological complexity and the environment *differs across environments*, we present further analysis (specifically in relation to experiment MO). Figure 5, illustrates the mean morphological simplicity over generations for both experiment MO and SO in each environment. We observe that experiment MO produces nearly indistinguishable selection pressure for morphological simplicity across all three environments. It is also noted that experiment SO tends to maintain a bias towards morphological complexity over evolutionary time which further supports hypothesis 0 and related works [4]. Figure 6 contributes further insight into morphological simplicity, in relation to experiment MO, by examining the average of the four selection points (discussed in section 5.1) across the three environments. The figure indicates little to no correlation

	Environment 1		Environment 2		Environment 3	
Comparing experiment SO and	Task Performance	Morphological Simplicity	Task Performance	Morphological Simplicity	Task Performance	Morphological Simplicity
$\bar{X}$	SO: [5, 10]	MO: [5, 10]	SO: [5, 10]	MO: [10, 20]	SO: [1, 5]	MO: [20, 30]
$\vee X$	SO: [0, 1]	MO: [0, 5]	=	MO: [0, 5]	=	MO: [5, 10]
X_ϵ	SO: [5, 10]	MO: [0, 5]	SO: [1, 5]	MO: [5, 10]	=	MO: [10, 20]
X_{KP}	SO: [0, 1]	MO: [10, 20]	SO: [1, 5]	MO: [10, 20]	=	MO: [10, 20]

Table 4: The superior performing solutions (SO or MO) and subsequent mean difference as a percentage (p<0.05) between SO solutions and each Pareto selection point. This is calculated for both objectives across the three environments. Note: '=' means there was no mean difference between SO and the selection point.

between environments and morphological simplicity. Moreover, it can be seen that each selection point correlates differently to each environment. $\bar{X}$ and X_ϵ maintain consistent morphological simplicity across environments while $\vee X$ and X_{KP} seem to display a concave and convex relationship respectfully across environments.

These results validate hypothesis 1 and show that there exists no direct trend between increasing environmental complexity and increased morphological complexity for MRS. We present two propositions for these findings.

(1) Other works report that evolving different morphological components (mechanical [5], triangular mesh [4]) induce orthogonal morphological trends to increasing environmental complexity. However, these works (specifically evolving single-robot systems) report some sort of trend where our work does not. We suspect that this is due to nature of MRS who tend to favour simplicity in order to evolve emergent cooperative behaviour [6, 7, 19, 49]. Moreover, related work on MRS have reported similar findings that are inline with this hypothesis [27, 61, 62]. Such a finding could prove promising for the general fields of MRS [19] and swarm robotics [7] where cheap sensory configurations are often a critical objective across a range of dynamic task environments [49].

(2) We theorize that such results are due to increased controller complexity. More specifically that evolutionary mechanisms react to the imposition of a cost on morphological complexity by increasing controller complexity. Figure 5 alludes to this as morphological simplicity remains somewhat constant from generation 50 on-wards. Such a theory is aligned with embodied cognition which states that intelligent behaviour is derived from the relationship between controller, morphology and environment [45, 46]. However, the substantiation of such a theory requires further experiments and thus these results are preliminary.

5.3 Future Works

Future works aims to further investigate the validity of the two propositions stated for hypothesis 1. For proposition 1, future works should investigate the cost of complexity for other collective gathering tasks such as collective construction as well as for single-robot systems. Thus giving further insight into the relationship between MRS and task complexity. Proposition 2 requires further analysis on controller complexity in relation to this study. An extension of this could also review the inverse, namely the effects on morphological evolution when a cost is imposed on *controller complexity*. We also suggest future extensions exploring the relationship between task performance and a cost on complexity for both controller and morphology. This paper also presented a novel method for evolving controller and morphology *HyperNEAT-M*. While similar works have showed the effectiveness of HyperNEAT-M's core procedure [48], future works should further investigate this method against other comparative co-evolution mechanisms. In relation to the multi-objective implementation used in this paper, a future works should compare this implementation with other multi-objective neuro-evolution methods such as the more widely used genotypic diversity function.

6 CONCLUSIONS

This research presented an initial contribution to the advantages of imposing a cost on morphological (sensory) complexity for the indirect-encoded co-evolution of controller and morphology for MRS. More specifically, this study compared the impact on both task performance and morphological evolution when there is a cost and no cost imposed on morphological complexity in various environments. Task performance is formalized as the ability to perform a collective gathering task and environments differed in the varying degrees of cooperation required to solve the task. Results indicated that the imposition of a cost on morphological complexity leads to cheaper morphologies with negligible difference in task performance across all environments. It was additionally found that increasing task complexity does not require an increase in morphological complexity. Accordingly, this study contributes two main findings to the related field of MRS and ER. Firstly, constraining complexity can result in more economical sensory configurations. Secondly, MRS designs do not require increasingly complex morphologies for more difficult task. This result has important implications for evolving MRS in dynamic environments that vary in difficulty. This work also offers preliminary insight into the relationship between morphology and environment for MRS.

7 ACKNOWLEDGEMENTS

We gratefully acknowledge the guidance and support received by our supervisor Dr. Geoff Nitschke. We would also like to thank Alexander Furman for his assistant and collaboration as a research partner. Computations were performed using facilities provided by the Centre of High Performance computing: chpc.ac.za. and the University of Cape Town's ICTS High Performance Computing team: hpc.uct.ac.za.

REFERENCES

[1] H. Abbass, R. Sarker, and C. Newton. Pde: a pareto-frontier differential evolution approach for multi-objective optimization problems. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, volume 2, pages 971–978. IEEE, 2001.

[2] O. Abramovich and A. Moshaiov. Multi-objective topology and weight evolution of neuro-controllers. In *Evolutionary Computation (CEC), 2016 IEEE Congress on*, pages 670–677. IEEE, 2016.

[3] J. Auerbach. *The Evolution of Complexity in Autonomous Robots.* PhD thesis, The University of Vermont, 2013.

[4] J. Auerbach and J. Bongard. On the relationship between environmental and morphological complexity in evolved robots. In *Proceedings of the 14th annual conference on Genetic and evolutionary computation*, pages 521–528. ACM, 2012.

[5] J. E. Auerbach and J. C. Bongard. On the relationship between environmental and mechanical complexity in evolved robots. In *Artificial Life 13*, number EPFL-CONF-191280, pages 309–316. MIT Press, 2012.

[6] G. Baldassarre, S. Nolfi, and D. Parisi. Evolving mobile robots able to display collective behaviors. *Artificial life*, 9(3):255–267, 2003.

[7] G. Beni. From swarm intelligence to swarm robotics. In *International Workshop on Swarm Robotics*, pages 1–9. Springer, 2004.

[8] X. Blasco, J. Herrero, J. Sanchis, and M. Martínez. A new graphical visualization of n-dimensional pareto front for decision-making in multiobjective optimization. *Information Sciences*, 178(20):3908–3924, 2008.

[9] J. Bongard. The utility of evolving simulated robot morphology increases with task complexity for object manipulation. *Artificial life*, 16(3):201–223, 2010.

[10] M. Branke. Opinion makers section. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[11] M. Bugajska and A. Schultz. Co-evolution of form and function in the design of autonomous agents: Micro air vehicle project. Technical report, Naval Research Lab Washington DC Center for Applied Research in Artificial Intelligence, 2000.

[12] N. Cheney, J. Bongard, and H. Lipson. Evolving soft robots in tight spaces. In *Proceedings of the 2015 annual conference on Genetic and Evolutionary Computation*, pages 935–942. ACM, 2015.

[13] N. Cheney, J. Clune, and H. Lipson. Evolved electrophysiological soft robots. In *ALIFE*, volume 14, pages 222–229, 2014.

[14] C. Coello. Evolutionary multi-objective optimization: a historical view of the field. *IEEE computational intelligence magazine*, 1(1):28–36, 2006.

[15] K. Deb and S. Gupta. Understanding knee points in bicriteria problems and their implications as preferred solution principles. *Engineering optimization*, 43(11):1175–1204, 2011.

[16] S. Didi and G. Nitschke. Neuro-evolution for multi-agent policy transfer in robocup keep-away. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 1281–1282. International Foundation for Autonomous Agents and Multiagent Systems, 2016.

[17] S. Doncieux, N. Bredeche, J. Mouret, and A. Eiben. Evolutionary robotics: what, why, and where to. *Frontiers in Robotics and AI*, 2:4, 2015.

[18] S. Doncieux and J. Mouret. Beyond black-box optimization: a review of selective pressures for evolutionary robotics. *Evolutionary Intelligence*, 7(2):71–93, 2014.

[19] A. Eiben, G. Nitschke, and M. Schut. Collective specialization for evolutionary design of a multi-robot system. In *International Workshop on Swarm Robotics*, pages 189–205. Springer, 2006.

[20] A. Eiben and M. Schoenauer. Evolutionary computing. *Information Processing Letters*, 82(1):1–6, 2002.

[21] J. Ferreira, C. Fonseca, and A. Gaspar. Methodology to select solutions from the pareto-optimal set: a comparative study. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 789–796. ACM, 2007.

[22] R. Fisher. *The genetical theory of natural selection: a complete variorum edition.* Oxford University Press, 1999.

[23] D. Floreano, P. Durr, and C. Mattiussi. Neuroevolution: from architectures to learning. *Evolutionary Intelligence*, 1(1):47–62, 2008.

[24] D. Floreano and F. Mondada. Automatic creation of an autonomous agent. In *From Animals to Animats 3: Proceedings of the Third International Conference on Simulation of Adaptive Behavior*, pages 421–430. The MIT Press, 1994.

[25] F. Gomez. *Robust non-linear control through neuroevolution.* PhD thesis, 2003.

[26] N. Hazon and G. Kaminka. Redundancy, efficiency and robustness in multi-robot coverage. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 735–741. IEEE, 2005.

[27] J. Hewland and G. Nitschke. The benefits of adaptive behavior and morphology for cooperation. In *Computational Intelligence, 2015 IEEE Symposium Series on*, pages 1047–1054. IEEE, 2015.

[28] P. Krcah. Evolving virtual creatures revisited. In *GECCO*, volume 7, pages 341–341, 2007.

[29] F. Lambercy and J. Tharin. *Khepera III User Manual: Version 3.5.* K-Team Corporation, Lausanne, Switzerland, 2013.

[30] J. Lehman, S. Risi, D. D'Ambrosio, and K. Stanley. Encouraging reactivity to create robust machines. *Adaptive Behavior*, 21(6):484–500, 2013.

[31] J. Lehman, K. Stanley, and R. Miikkulainen. Effective diversity maintenance in deceptive domains. In *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, pages 215–222. ACM, 2013.

[32] R. Lenski, C. Ofria, R. Pennock, and C. Adami. The evolutionary origin of complex features. *Nature*, 423(6936):139, 2003.

[33] W. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[34] R. Miikkulainen, E. Feasley, L. Johnson, I. Karpov, P. Rajagopalan, A. Rawal, and W. Tansey. *Multiagent Learning through Neuroevolution*, pages 24–46. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[35] D. Moriarty and R. Miikkulainen. Forming neural networks through efficient and adaptive coevolution. *Evolutionary Computation*, 5(4):373–399, 1997.

[36] R. Murphy, S. Tadokoro, D. Nardi, A. Jacoff, P. Fiorini, H. Choset, and A. Erkmen. Search and rescue robotics. In *Springer Handbook of Robotics*, pages 1151–1173. Springer, 2008.

[37] G. Nitschke. Co-evolution of cooperation in a pursuit evasion game. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 2, pages 2037–2042. IEEE, 2003.

[38] G. Nitschke. Neuro-evolution for emergent specialization in collective behavior systems. 2009.

[39] G. Nitschke, M. Schut, and A. Eiben. Emergent specialization in biologically inspired collective behavior systems. In *Intelligent complex adaptive systems*, pages 215–253. IGI Global, 2008.

[40] S. Nolfi, D. Floreano, O. Miglino, and F. Mondada. How to evolve autonomous robots. In *Artificial Life IV: Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*, pages 190–197. MIT Press, 1994.

[41] S. Nolfi and D. Parisi. Evolving non-trivial behaviors on real robots: an autonomous robot that picks up objects. In *Congress of the Italian Association for Artificial Intelligence*, pages 243–254. Springer, 1995.

[42] R. Odagiri, W. Yu, T. Asai, O. Yamakawat, and K. Murase. Measuring the complexity of the real environment with evolutionary robot: Evolution of a real mobile robot khepera to have a minimal structure. In *Evolutionary Computation Proceedings, 1998.*, pages 348–353. IEEE, 1998.

[43] H. Orr. Adaptation and the cost of complexity. *Evolution*, 54(1):13–20, 2000.

[44] L. Parker. Alliance: An architecture for fault tolerant multirobot cooperation. *IEEE transactions on robotics and automation*, 14(2):220–240, 1998.

[45] R. Pfeifer and J. Bongard. *How the body shapes the way we think: a new view of intelligence.* MIT press, 2006.

[46] R. Pfeifer and C. Scheier. *Understanding intelligence.* MIT press, 2001.

[47] R. Putter and G. Nitschke. Evolving morphological robustness for collective robotics. In *Computational Intelligence (SSCI), 2017 IEEE Symposium Series on*, pages 1–8. IEEE, 2017.

[48] S. Risi and K. Stanley. A unified approach to evolving plasticity and neural geometry. In *Neural Networks (IJCNN), The 2012 International Joint Conference on*, pages 1–8. IEEE, 2012.

[49] M. Rubenstein, A. Cornejo, and R. Nagpal. Programmable self-assembly in a thousand-robot swarm. *Science*, 345(6198):795–799, 2014.

[50] S. Russell and P. Norvig. *Artificial intelligence: a modern approach.* Malaysia; Pearson Education Limited,, 2016.

[51] T. Salimans, J. Ho, X. Chen, and I. Sutskever. Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint*, 2017.

[52] J. Schrum and R. Miikkulainen. Evolving multimodal behavior with modular neural networks in ms. pac-man, 2014.

[53] K. Sims. Evolving 3d morphology and behavior by competition. *Artificial life*, 1(4):353–372, 1994.

[54] K. Sims. Evolving virtual creatures. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 15–22. ACM, 1994.

[55] T. Smith. Adding vision to khepera: An autonomous robot footballer. *Master's thesis, School of Cognitive and Computing Sciences, University of Sussex*, 1997.

[56] N. Srinivas and K. Deb. Muiltiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary computation*, 2(3):221–248, 1994.

[57] K. Stanley. Compositional pattern producing networks: A novel abstraction of development. *Genetic programming and evolvable machines*, 8(2):131–162, 2007.

[58] K. Stanley, D. D'Ambrosio, and J. Gauci. A hypercube-based encoding for evolving large-scale neural networks. *Artificial life*, 15(2):185–212, 2009.

[59] K. Stanley and R. Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary computation*, 10(2):99–127, 2002.

[60] V. Trianni and M. López-Ibáñez. Advantages of task-specific multi-objective optimisation in evolutionary robotics. *PloS one*, 10(8):e0136406, 2015.

[61] J. Watson and G. Nitschke. Deriving minimal sensory configurations for evolved cooperative robot teams. In *Evolutionary Computation (CEC), 2015 IEEE Congress on*, pages 3065–3071. IEEE, 2015.

[62] J. Watson and G. Nitschke. Evolving robust robot team morphologies for collective construction. In *Computational Intelligence, 2015 IEEE Symposium Series on*, pages 1039–1046. IEEE, 2015.

[63] W. Willigen, E. Haasdijk, and L. Kester. A multi-objective approach to evolving platooning strategies in intelligent transportation systems. In *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, pages 1397–1404. ACM, 2013.

[64] X. Yao. Evolving artificial neural networks. *Proceedings of the IEEE*, 87(9):1423–1447, 1999.

[65] J. Zhao and G. Peng. Neat versus pso for evolving autonomous multi-agents coordination on pursuit-evasion problem. In *Informatics in Control, Automation and Robotics*, pages 711–717. Springer, 2011.

[66] E. Zitzler, M. Laumanns, and L. Thiele. Spea2: Improving the strength pareto evolutionary algorithm. *TIK-report*, 103, 2001.