Repositório ISCTE-IUL

Cooperative Coevolution of Control for a Real Multirobot System

Jorge Gomes[1,2,3], Miguel Duarte[1,2,4], Pedro Mariano[3], and
Anders Lyhne Christensen[1,2,4]

[1]BioMachines Lab, Lisbon, Portugal
[2]Instituto de Telecomunicações, Lisbon, Portugal
[3]BioISI, Faculdade de Ciências da Universidade de Lisboa, Portugal
[4]Instituto Universitário de Lisboa (ISCTE-IUL), Lisbon, Portugal
`jgomes@di.fc.ul.pt`

Abstract. The potential of cooperative coevolutionary algorithms (CCEAs) as a tool for evolving control for heterogeneous multirobot teams has been shown in several previous works. The vast majority of these works have, however, been confined to simulation-based experiments. In this paper, we present one of the first demonstrations of a real multirobot system, operating outside laboratory conditions, with controllers synthesised by CCEAs. We evolve control for an aquatic multirobot system that has to perform a cooperative predator-prey pursuit task. The evolved controllers are transferred to real hardware, and their performance is assessed in a non-controlled outdoor environment. Two approaches are used to evolve control: a standard fitness-driven CCEA, and novelty-driven coevolution. We find that both approaches are able to evolve teams that transfer successfully to the real robots. Novelty-driven coevolution is able to evolve a broad range of successful team behaviours, which we test on the real multirobot system.

Keywords: Cooperative coevolution, evolutionary robotics, novelty search, reality gap, heterogeneous multirobot systems

1 Introduction

Cooperative coevolutionary algorithms (CCEAs) allow for the evolution of solutions that consist of coadapted, interacting components [16, 17]. CCEAs are a natural fit for the evolution of heterogeneous multiagent systems [18], as each agent can be represented as an independent component of the solution, and can therefore evolve a specialised behaviour (see for instance [12, 18, 22]). The classic CCEA architecture [17] operates with two or more populations, where each agent evolves in a separate population. Populations are isolated from one another, meaning that individuals only compete and reproduce with members of their own population. The individuals in each population are evaluated by forming teams with representative individuals from the other populations. These teams are evaluated in the problem domain, and the individual under evaluation receives the fitness score obtained by the team as a whole.

Previous works that have applied CCEAs to the evolution of agent behaviours can be divided in three main categories [14]: (i) *game-theoretic environments*, essentially strategy games where each agent is rewarded according to a payoff matrix [15, 21]; (ii) *abstract embodied agents*, where the evolved agents are situated in an environment that they sense and act in, but the agents are abstract and unrelated to any real robotic platform [7, 18, 22]; and (iii) *simulated robotics tasks*, in which the evolved agents are modelled closely after a real robotic platform and a real task environment [10, 11].

One notable category is missing from this list, namely *real robotics tasks* – tasks in which behavioural control is evolved in simulation, and then transferred to a real robot team. While this reality gap has been crossed using other evolutionary algorithms [19], in both single [8] and multirobot systems [3], to the best of our knowledge, CCEAs have been confined to simulation-based experiments up until now. The potential of CCEAs to evolve control for robot teams has been shown in simulation in tasks such as: predator-prey pursuit [7, 11], herding [18], collective construction [13], multirobot foraging [5, 12], and keepaway soccer [4].

In this paper, we evolve control for an aquatic surface multirobot system that must perform a cooperative predator-prey pursuit task. Predator-prey pursuit is one of the most commonly studied tasks in multiagent coevolution. In the cooperative version of this task [7, 11, 22], a team of predators must cooperate to capture an escaping prey. The predator-prey task is especially interesting in CCEA studies because behavioural heterogeneity and close cooperation in the predator team is required to effectively catch the prey [22]. After evolving the controllers offline in simulation, we transfer the controllers to the real robotic platform, and systematically evaluate them in an outdoor environment. The natural unpredictability associated with the aquatic environment (caused by inaccurate robot motion, waves, and currents) allow us to study transferability in a realistic scenario, and understand how controllers evolved by CCEAs are able to cope with noisy and stochastic conditions.

We evolve control using two cooperative coevolution approaches: a standard fitness-driven CCEA [17], and novelty-driven cooperative coevolution [7] – a recently proposed algorithm that aims at mitigating the premature convergence issues that commonly plague CCEAs [15,16]. Novelty-driven coevolution is based on novelty search [9], an evolutionary approach that rewards individuals displaying novel behaviours, rather than exclusively rewarding the individuals that display the highest performance with respect to a fitness function. Novelty-driven coevolution (*NS-Team*) relies on team-level behaviour characterisations, and rewards behaviourally novel teams in addition to high-fitness ones, as it is typically done in CCEAs. The team-level characterisations capture how the team as a whole behaves, without discriminating between the behaviours of the individual agents. Both the fitness and the novelty scores of the teams are used to reward the individuals, via a multi-objective algorithm. By rewarding agents that lead to novel team behaviours, an evolutionary pressure towards novel equilibrium states is created. Besides the ability to overcome premature convergence, and thus reach higher quality solutions, it has also been shown that *NS-Team* can evolve a diverse set of solutions for a given task [4, 5, 7].

2 Experimental Setup

2.1 Cooperative Predator-prey Task

In our predator-prey pursuit task, a team of three predators must cooperate to capture one escaping prey. Only the controllers of the team of predators are evolved, while the prey has a pre-specified fixed behaviour.
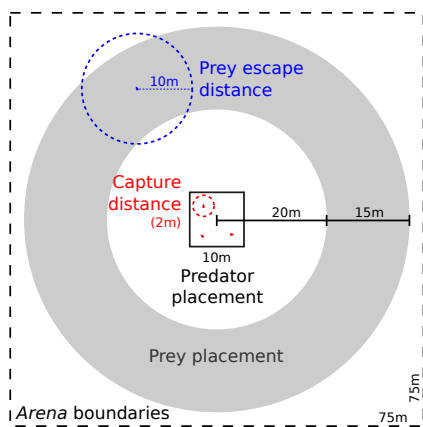

Fig. 1. Task setup used for the evolutionary process.

In each trial, the three predators are placed in the centre of the arena, with random positions and orientations (Figure 1). The prey is placed in a random location, ranging from 20 m to 35 m from the centre of the arena. A trial ends if a predator gets closer than 2 m from the prey (prey is *captured*), if the prey escapes the arena, or if the time limit (75 s) is reached. The prey moves in the opposite direction of the closest predator, if that predator is closer than 10 m, otherwise it does not move. The prey can move up to the maximum possible speed of the predators, meaning that the predators typically cannot outrun it. Cooperation among the predators is therefore essential to capture the prey.

2.2 Robotic Platform

For our experiments, we use an aquatic multirobot system [1] that has been used in other evolutionary robotics studies in the past [3]. Each robot is a small (65 cm in length) differential drive mono-hull robot. The robots can move at speeds of up to 1.7 m/s, turn at a maximum rate of 90°/s, and are equipped with GPS and compass. The robots broadcast information (such as their position) to their neighbours up to a range of 40 m using Wi-Fi, which is then used to calculate the robots' sensory inputs. The same robotic platform is used for both the predator robots and prey robot.

Each predator is controlled by an artificial neural network, which receives the sensory inputs – the distance to other predators and the relative position of the prey – and has two outputs that control the linear speed and the angular velocity of the robot. The two output values are converted to left and right motor speeds and applied to the robot's motors. The network relies on the following sensory inputs, which are limited to a range of 40 m, and are normalised to $[-1, 1]$:

Predator sensing: Six inputs for detecting the other predators, corresponding to six equally-sized circular sectors around the robot. Each input returns the normalised distance to the closest predator in the corresponding sector, or the maximum value if no predator is present there.

Prey location: Two inputs returning (i) the relative angle from the predator to the prey (zero corresponds to straight ahead), and (ii) the normalised distance from the predator to the prey. If the prey is not within sensing range, the sensors return an angle of zero and the maximum distance.

2.3 Evolutionary Setup

Both fitness-driven and novelty-driven cooperative coevolution were implemented over the same standard coevolutionary architecture [17]. There are three coevolving populations, one for each of the predators. Every generation, each population is evaluated in turn. To evaluate an individual from one population, a team is formed with one representative from each other population – the individual that obtained the highest fitness score in the previous generation, or a random one in the first generation. Only the individual currently under evaluation receives the score obtained by the team. Every team is evaluated in 10 simulation trials, with randomized initial conditions. The controllers of each population are evolved by NEAT [20], a neuroevolution algorithm extensively used in evolutionary robotics, that evolves both the weights and topology of the networks. The three coevolving populations use the parameters listed in Table 1.

The fitness function F is the same as the one used in [7], which rewards the teams for capturing the prey as soon as possible, or getting close to it:

$$F = \begin{cases} 2 - \tau/T & \text{if prey captured} \\ max(0, (d_i - d_f)/size) & \text{otherwise} \end{cases}, \tag{1}$$

where τ is the time to capture the prey, T is the maximum trial length, d_i and d_f are, respectively, the average initial and final distance from the predators to the prey, and $size$ is the side length of the arena.

Novelty-driven coevolution is implemented as proposed in [7], using the *NS-Team* technique, which computes the individuals' novelty scores based on the behavioural novelty displayed by the team in which the individual participated. To calculate the novelty score of each team, we rely on four features to characterise team behaviour, all normalised to [0,1]: (i) whether the prey was captured or not; (ii) average final distance of the predators to the prey; (iii) average distance of each predator to the other predators over the trial; and (iv) trial length. The novelty search algorithm was configured according to [6], see Table 1 (last row). The novelty score of each individual is combined with its fitness score using the NSGA-II [2] multiobjective ranking, as advocated in previous works [6,7].

Table 1. Parameters used for NEAT and novelty search (last row).

Population size	150	Target species count	5	Crossover prob.	20%
Recurrency allowed	true	Mutation prob.	25%	Prob. add link	5%
Prob. add node	3%	Prob. mutate bias	30%	Num. generations	250
Novelty k-nearest	15	Add archive prob.	2.5%	Max. archive size	2000

2.4 Simulation

For the evolutionary process, we used a two-dimensional simulation environment, where the robots are abstracted as circular objects with a certain heading and position[1]. The robot motion model was implemented based on simple measurements taken on the real robots, and did not include complex physics simulation or fluid dynamics. In order to facilitate the transfer from simulation to reality and promote general behaviours, noise was applied to the sensors and actuators [3, 8] based on measurements taken from real robots, and the initial task conditions were varied in every simulation trial. The following parameters were varied during the simulated trials:

Set for each trial: random individual motor speed offsets of up to 10% of maximum speed; compass offset up to $\pm 9°$; the prey's escape speed varied between [75%,100%] of the predators's maximum speed; and the initial positions and orientations of all robots were varied according to Section 2.1.

Set at each time step: GPS error up to $1.8\,\text{m}$; compass error up to $\pm 10°$; motor output varied up to 5%; and the prey's escape direction randomly varied up to 50% from the optimal direction.

3 Evolving and Identifying Diverse Behaviours

For both fitness-driven (*Fit*) and novelty-driven cooperative coevolution (*NS-Team*), we followed a methodology that allowed us to identify a set of diverse and high-quality solutions, that were then evaluated in the real multirobot system.

Evolutionary process: Each evolutionary approach was repeated in ten independent evolutionary runs. To obtain a more accurate estimate of the evolved teams quality and behaviour, all the *best-of-generation* teams (the teams that obtained the highest fitness score in each generation, in each evolutionary run) were re-evaluated a posteriori in 50 simulation trials. On average, the evolutionary runs of *Fit* achieved a highest fitness score of 1.09 ± 0.10, and *NS-Team* achieved 0.96 ± 0.20. While this difference is significant ($p = 0.043$, Mann-Whitney U test), both approaches managed to evolve high-quality solutions.

Behaviour mapping: To visualise the diversity of behaviours evolved by each evolutionary approach, we mapped the *best-of-generation* teams according to their behaviour characterisation vector (see Section 2.3), as done in previous works [7]. The four dimensions of the behaviour characterisation were reduced to two dimensions using a Kohonen self-organising map in order to obtain a visual representation of the team behaviour space exploration, see Figure 2 (left). The teams evolved by each evolutionary approach were then mapped: each team is assigned to the node (map region) with the closest weight vector, see Figure 2. The results show that *NS-Team* explored the behaviour space much more uniformly, and could reach behaviour regions that were never reached by *Fit*, which is consistent with the results reported in previous works [7].
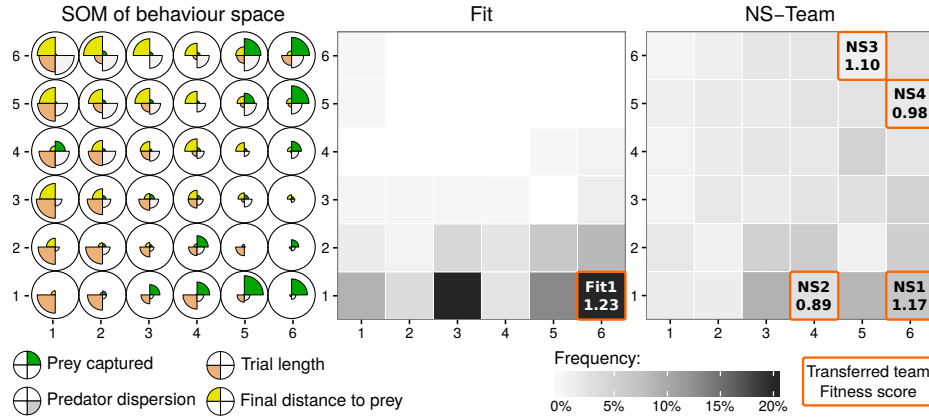
---

[1] https://github.com/BioMachinesLab/drones/tree/master/JBotAquatic

Fig. 2. Left: trained Kohonen map, where each node represents a region of the team behaviour space. Middle and right: team behaviour exploration by the two evolutionary approaches. The darker a region, the more of the evolved teams belonged to it.

Selection of solutions: We then proceeded to select a diverse set of solutions to be tested in the real robots. We selected different regions of the behaviour space where the prey capture rate was high, and identified the team belonging to each of those regions that obtained the highest fitness score, see Figure 2 (bottom). We chose one team evolved by *Fit*, as all the high-quality teams were found in the bottom-right corner of the map, and four solutions evolved by *NS-Team*, from different regions of the map with high *prey capture* values.

4 Transferring the Teams to Real Robots

The selected teams were then evaluated in the real multirobot system. The experiments were performed in a semi-enclosed water body, see Figure 3. The task setup was similar to the simulation setup (see Section 2.1): the three predator robots were placed close to the centre of the arena, and the prey was placed at approximately 25, 30, and 35 m away from the centre, in each of the three trials that were used to assess the performance of the teams. Each trial lasted for at most 100 s, the arena boundaries were 100×100 m, and the prey moved at the maximum speed. To compare the results of the real-robot experiments with simulation, the chosen teams were re-evaluated in 500 simulation trials, using the same initial conditions as the real-robot experiments. The fitness scores and behaviour features of the teams operating in the real environment were computed using logged GPS data.

In Figure 4 (Fitness), we compare the fitness scores obtained by the teams in simulation and in the real robots. We additionally explore the diversity of team behaviours by comparing the controllers' performance in reality and in simulation according to the behaviour features that were used in novelty-driven coevolution (Section 2.3). The results show that all teams except *NS2* were able
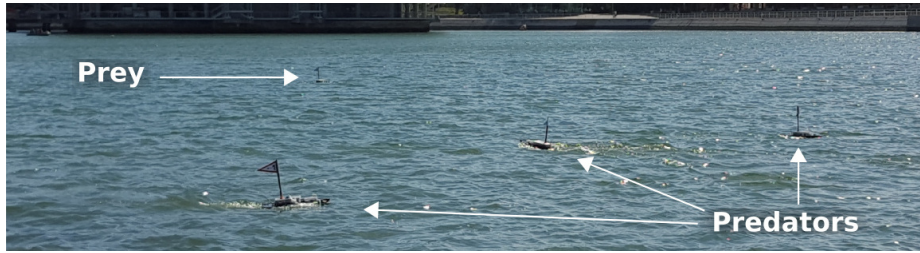
Fig. 3. Photo of the real-robot experiments, at Parque das Nações, Lisbon, Portugal, in a semi-enclosed area in the margin of the Tagus river.
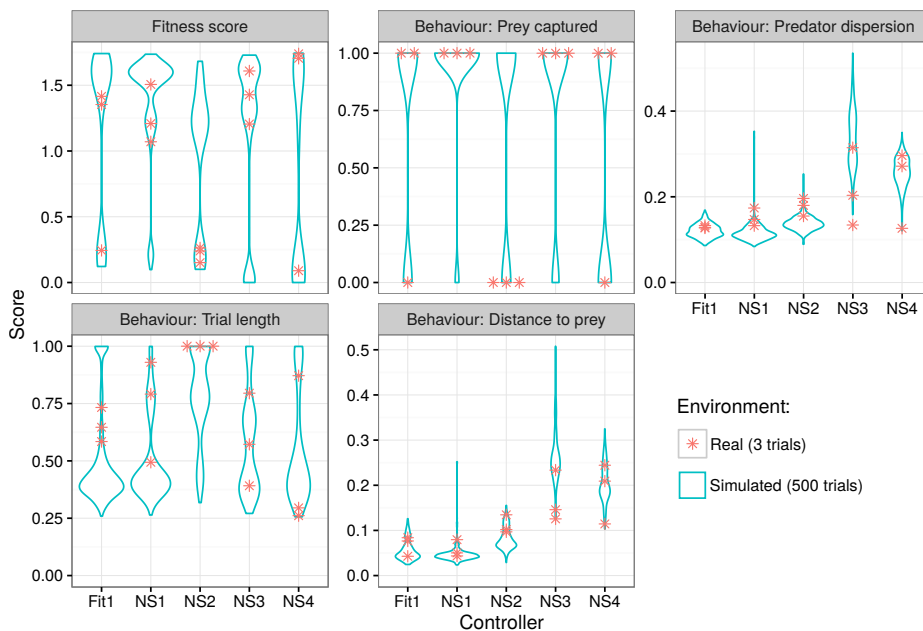

Fig. 4. Comparison of the fitness score and behaviour features obtained in the real-robot experiments (asterisks) and in simulation (violin plots) in similar conditions.

to capture the prey in the majority of the trials. The fitness scores obtained in the real experiments are similar to the scores obtained in simulation, fitting in the distribution obtained in simulation. These results are a first indication that the evolved controllers were generally able to cross the reality gap successfully.

The effectiveness of the team behaviours was confirmed by analysing the traces of the real-robot experiments, shown in Figure 5. The *Fit1* and *NS1* teams displayed a behaviour where the three predators would initially spread and move towards the prey, each approaching the prey from a different direction. The behaviour of *NS2* was similar to *Fit1* and *NS1*, but the predator team dispersed more. The teams *NS3* and *NS4* displayed a significantly different behaviour:
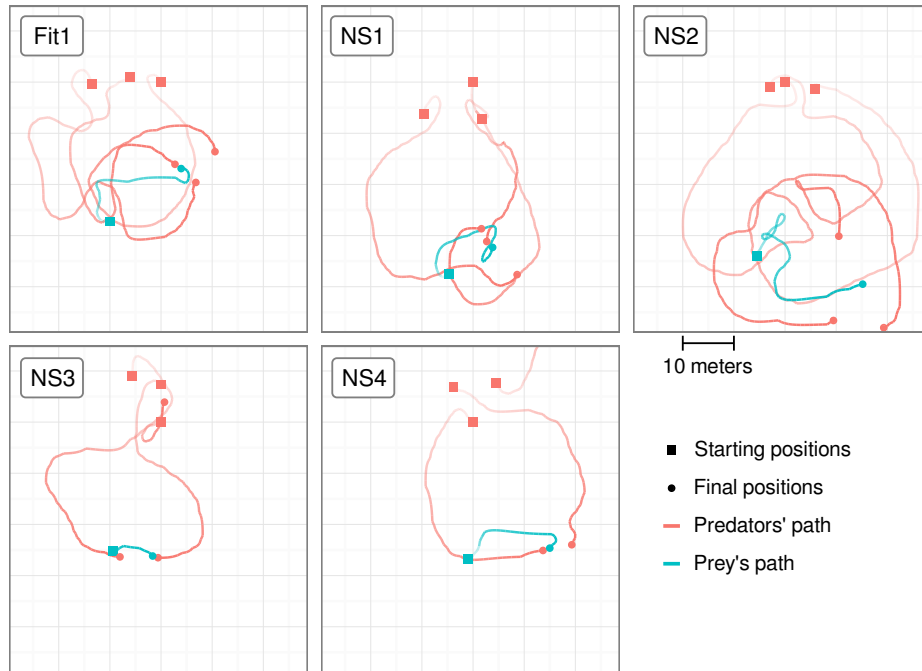
Fig. 5. Traces of one experimental trial (out of three) for each of the teams evaluated in the real robots. Traces and videos of all real-robot experiments are available online.[2]

only two predators chased the prey, approaching it from opposite directions, while the remaining predator would move away from the group. The observed robot traces are consistent with the measured behavioural features (Figure 4), and confirm that novelty-driven coevolution was able to achieve a wide diversity of team behaviours. For instance, it is possible to observe that *NS3* and *NS4* display a higher dispersion and final distance to prey, which is explained by the fact that in these teams, only two predators chase the prey. The differences and similarities between the team behaviours observed in the real-robot experiments are consistent with the behaviour map obtained in simulation (Figure 2).

Overall, despite the stochastic conditions of the aquatic environment, the predators displayed effective cooperation, and were consistently able to solve the task. The team of predators would often fail to capture the prey in the first attempt, but the team would then spread out and try to encircle the prey again. Moreover, robots sometimes displayed temporary motor failures (see supplementary videos[2]), which did not compromise the effectiveness of the team. These behaviours suggest that the teams were not overfitted to the simulation environment, and could effectively adapt to different scenarios.

---

[2] Videos and logs of the experiments: `http://dx.doi.org/10.5281/zenodo.49582`

5 Conclusion

In this paper, we employed cooperative coevolutionary algorithms (CCEAs) to evolve control for an aquatic multirobot system. Our experiments relied on a cooperative predator-prey task, where a heterogeneous team of three predators was evolved to capture one reactive prey. Two evolutionary approaches were applied: traditional fitness-driven cooperative coevolution, and novelty-driven cooperative coevolution. The evolutionary processes were conducted exclusively in simulation, and a number of high-fitness teams were then systematically evaluated in real robots operating in a non-controlled outdoor environment.

The evolved teams generally transferred well to the real robots, successfully crossing the reality gap. Out of the five teams tested, four teams could consistently capture the prey, and obtained fitness scores very similar to those obtained in simulation. The cooperation between robots that was exhibited in simulation was also observed in real robots, and the teams displayed robust behaviours that did not appear to be overfitted to the simulation environment. The successful transfer is especially notable given that we used low-fidelity simulator during evolution, and given the stochastic nature of the real task environment. We encouraged the evolution of robust and transferable controllers by introducing conservative amounts of noise and variations in the sensors and actuators of the robots in simulation, and by using multiple trials to evaluate each solution, with different initial conditions.

Novelty-driven cooperative coevolution was able to produce a good diversity of high-quality team behaviours for solving the task, which were identified following a systematic approach. The diversity of behaviours that was observed in simulation was also present in the real multirobot system.

In summary, we demonstrated that CCEAs can be successfully used to synthesise control for a real multirobot system, operating in an environment outside controlled laboratory conditions. Despite the large number of previous works that have showed the potential of CCEAs for evolving heterogeneous multirobot systems, our work stands amongst the first to demonstrate this potential in real robots and in a realistic environment. Our experiments also validated, for the first time, the potential of novelty-driven cooperative coevolution in real robots, and confirmed it as a valuable approach to evolve diverse team behaviours.

References

1. Costa, V., Duarte, M., Rodrigues, T., Oliveira, S.M., Christensen, A.L.: Design and development of an inexpensive aquatic swarm robotics system. In: Proceedings of the MTS/IEEE OCEANS. IEEE Press, Piscataway, NJ (2016), in press
2. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002)

3. Duarte, M., Costa, V., Gomes, J., Rodrigues, T., Silva, F., Oliveira, S.M., Christensen, A.L.: Evolution of collective behaviors for a real swarm of aquatic surface robots. PLoS ONE 11(3), e0151834 (2016)
4. Gomes, J., Mariano, P., Christensen, A.L.: Avoiding convergence in cooperative coevolution with novelty search. In: International Conference on Autonomous Agents & Multiagent Systems (AAMAS). pp. 1149–1156. IFAAMAS (2014)
5. Gomes, J., Mariano, P., Christensen, A.L.: Cooperative coevolution of morphologically heterogeneous robots. In: European Conference on Artificial Life. pp. 312–319. MIT Press (2015)
6. Gomes, J., Mariano, P., Christensen, A.L.: Devising effective novelty search algorithms: A comprehensive empirical study. In: Genetic and Evolutionary Computation Conference (GECCO). pp. 943–950. ACM Press (2015)
7. Gomes, J., Mariano, P., Christensen, A.L.: Novelty-driven cooperative coevolution. Evolutionary Computation (2016), in press
8. Jakobi, N.: Evolutionary robotics and the radical envelope-of-noise hypothesis. Adaptive Behavior 6(2), 325–368 (1997)
9. Lehman, J., Stanley, K.O.: Abandoning objectives: Evolution through the search for novelty alone. Evolutionary Computation 19(2), 189–223 (2011)
10. Nitschke, G.: Designing emergent cooperation: a pursuit–evasion game case study. Artificial Life and Robotics 9(4), 222–233 (2005)
11. Nitschke, G.S., Eiben, A.E., Schut, M.C.: Evolving team behaviors with specialization. Genetic Programming and Evolvable Machines 13(4), 493–536 (2012)
12. Nitschke, G.S., Schut, M.C., Eiben, A.E.: Collective neuro-evolution for evolving specialized sensor resolutions in a multi-rover task. Evolutionary Intelligence 3(1), 13–29 (2010)
13. Nitschke, G.S., Schut, M.C., Eiben, A.E.: Evolving behavioral specialization in robot teams to solve a collective construction task. Swarm and Evolutionary Computation 2, 25–38 (2012)
14. Panait, L., Luke, S.: Cooperative multi-agent learning: The state of the art. Autonomous Agents & Multi-Agent Systems 11(3), 387–434 (2005)
15. Panait, L., Luke, S., Wiegand, R.P.: Biasing coevolutionary search for optimal multiagent behaviors. IEEE Transactions on Evolutionary Computation 10(6), 629–645 (2006)
16. Popovici, E., Bucci, A., Wiegand, R.P., De Jong, E.D.: Coevolutionary principles. In: Handbook of Natural Computing, pp. 987–1033. Springer (2012)
17. Potter, M.A., De Jong, K.A.: Cooperative coevolution: An architecture for evolving coadapted subcomponents. Evolutionary Computation 8(1), 1–29 (2000)
18. Potter, M.A., Meeden, L.A., Schultz, A.C.: Heterogeneity in the coevolved behaviors of mobile robots: The emergence of specialists. In: International Joint Conference on Artificial Intelligence (IJCAI). pp. 1337–1343. Morgan Kaufmann (2001)
19. Silva, F., Duarte, M., Correia, L., Oliveira, S.M., Christensen, A.L.: Open issues in evolutionary robotics. Evolutionary Computation 24(2), 205–236 (2016)
20. Stanley, K., Miikkulainen, R.: Evolving neural networks through augmenting topologies. Evolutionary Computation 10(2), 99–127 (2002)
21. Wiegand, R.P., Liles, W.C., De Jong, K.A.: Analyzing cooperative coevolution with evolutionary game theory. In: Congress on Evolutionary Computation (CEC). vol. 2, pp. 1600–1605. IEEE Press (2002)
22. Yong, C.H., Miikkulainen, R.: Coevolution of role-based cooperation in multiagent systems. IEEE Transactions on Autonomous Mental Development 1(3), 170–186 (2009)