Adaptation of Robot Behaviour through Online Evolution and Neuromodulated Learning

Fernando Silva[1], Paulo Urbano[1] and Anders Lyhne Christensen[2]

[1] LabMAg, Faculdade de Ciências, Universidade de Lisboa (FC-UL)
{fsilva,pub}@di.fc.ul.pt
[2] Instituto de Telecomunicações & Instituto Universitário de Lisboa (ISCTE-IUL)
anders.christensen@iscte.pt

Abstract. We propose and evaluate a novel approach to the online synthesis of neural controllers for autonomous robots. We combine online evolution of weights and network topology with neuromodulated learning. We demonstrate our method through a series of simulation-based experiments in which an e-puck-like robot must perform a dynamic concurrent foraging task. In this task, scattered food items periodically change their nutritive value or become poisonous. Our results show that when neuromodulated learning is employed, neural controllers are synthesised faster than by evolution alone. We demonstrate that the online evolutionary process is capable of generating controllers well adapted to the periodic task changes. An analysis of the evolved networks shows that they are characterised by specialised modulatory neurons that exclusively regulate the output neurons.

Keywords: Neural Networks, Online Adaptation, Neuroevolution, Neuromodulated Learning, odNEAT

1 Introduction

Evolutionary computation techniques have been widely studied and applied in the field of robotics as a means to automate the design of robotic systems [1]. In evolutionary robotics (ER), robot controllers are typically based on artificial neural networks (ANN). The connection weights and sometimes the topology of the ANN are optimised by an evolutionary algorithm (EA), a process termed as *neuroevolution*. Evolutionary synthesis of controllers is usually performed offline in simulation, which presents a number of limitations. When a suitable neurocontroller is found, it is deployed on real robots. Since no evolution or adaptation takes place online, the controllers are fixed solutions that remain static throughout the robot's lifetime. If environmental conditions or task parameters become distinct from those encountered during offline evolution, the evolved controllers may be incapable of solving the task as they have no means to adapt.

Online evolution is a process of continuous adaptation that potentially gives robots the capacity to respond to changes in the task or in environmental conditions by modifying their behaviour. An EA is executed on the robots themselves

as they perform their tasks. This way, robots may be capable of long-term self-adaptation. In recent years, different approaches to online evolution have been proposed (see for instance [2–4]). Notwithstanding, in such contributions, online neuroevolution has been limited to evolving weights in fixed-topology ANNs. In a recent study [9], we proposed a novel approach called odNEAT. odNEAT is an online, distributed, and decentralised EA for online evolution in groups of robots, that evolves both weights and network topology. The network topology is therefore a product of a continuous evolutionary process.

Online evolution is a form of online adaptation that acts at genotype level. Controllers produced are static as they do not change their parameters *while* they are controlling the robot. While evolution produces phylogenetic adaptation, online learning operates on a much shorter time-scale. Learning acts at phenotypic level and gives each individual controller the capacity to self-adjust during task-execution. Several studies indicate that learning can accelerate the evolution of good solutions, a phenomenon known as the Baldwin effect [5].

Agents controlled by ANNs can learn from experience by dynamically changing their internal synaptic strengths. This mechanism is inspired by how organisms in nature adapt to cope with dynamic and unstructured environments as a result of synaptic plasticity [13]. In this paper, we synthesise behavioural control for autonomous robots based on online evolutionary computation and online learning. We combine evolution of weights and network topology (odNEAT) with *neuromodulation* [12]. Neuromodulation is a form of synaptic modification involving modulatory neurons that diffuse chemicals at target synapses. Modulation has been suggested as essential for stabilising classical Hebbian plasticity and memory [15].

We demonstrate our method in a simulated experiment where an e-puck-like robot [8] must perform a dynamic concurrent foraging task. The robot must locate and consume scattered food items. When a food item is consumed, a new item of the same type is randomly placed in the environment. At regular time intervals, food items change their nutritive value, or become poisonous. Besides learning to forage, the robot must therefore be able to adapt and change its foraging policy in order to survive. To the best of our knowledge, the contribution presented here is the first demonstration of online learning and online evolution of both the weights and the ANN topology in multirobot systems.

2 Background

In this section, we first discuss evolution of plastic ANNs, with a focus on the neuromodulation-based model, and we then review odNEAT, which we extended to incorporate neuromodulated plasticity.

2.1 Artificial Evolution of Neuromodulated Plasticity

Synaptic plasticity is considered a fundamental mechanism behind memory and learning in biological neural networks [14]. In ANNs, the modification of internal synaptic connection strengths can be performed according to a generalised

Hebbian plasticity rule [13]. Synaptic weights are updated based on pre- and post-synaptic neuron activities as follows:

$$\Delta w = \eta \cdot [Axy + Bx + Cy + D], \tag{1}$$

where η is the learning rate, x and y are the activation levels of the pre-synaptic and post-synaptic neurons. w is the connection weight and $A - D$ are the correlation term, pre-synaptic term, post-synaptic term, and constant weight decay or increase, respectively. By tuning these parameters, it is possible to evolve distinct forms of synaptic plasticity. ANN controllers can thus implement learning and memory by means of recurrent connections, plastic Hebbian connections, or a combination of the two.

The adaptation capabilities of fixed-topology plastic Hebbian ANNs were demonstrated in [7]. In a light-switching task, a mobile robot Khepera had to turn on a light switch and then navigate towards a gray area at the opposite end of the environment. The evolved plastic Hebbian controllers managed to solve the task much faster than fixed-weight networks. The plastic controllers also exhibited a larger variety of successful behaviours and robustness to environmental changes. With a similar setup, it was shown that dynamic environments promote the genetic expression of plastic connections over static ones [6].

Although the use of plastic ANNs can increase performance, recent studies indicate that in more complex tasks, both plastic and fixed-weight ANNs have limited learning capabilities [11–13]. In this context, controlling synaptic plasticity through *neuromodulation* was presented as a more powerful and biologically plausible approach [14]. In biological neural networks, neuromodulation has been suggested as essential for stabilising classical Hebbian plasticity and memory [15]. In a neuromodulated network, specialised modulatory neurons control the amount of activity-dependent plasticity between pairs of standard control neurons. This process is illustrated in Fig. 1.

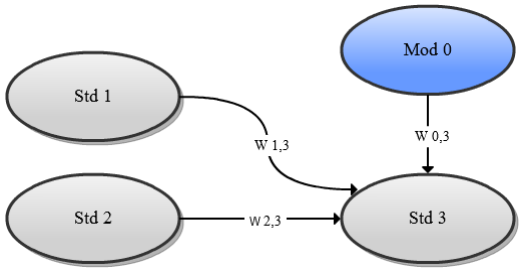

Fig. 1. Neuromodulated plasticity. A modulatory neuron, Mod 0, transmits a modulatory signal to Std 3. Modulation affects the learning rate for synaptic plasticity of weights $w_{1,3}$ and $w_{2,3}$. The weights are part of the incoming connections for the standard control neuron being modulated.

The advantage of adding neuromodulation is that ANNs become capable of changing the degree of synaptic plasticity on specific neurons at specific times, i.e., deciding when learning should start and stop. In addition to its standard activation value a_i, each neuron i also computes its modulatory activation m_i as follows:

$$a_i = \sum_{j \in Std} w_{ji} \cdot o_j, \tag{2}$$

$$m_i = \sum_{j \in Mod} w_{ji} \cdot o_j, \tag{3}$$

where w_{ji} is the connection weight between pre- and post-synaptic neurons j and i. o_j is the output of a pre-synaptic neuron j. The weight between neurons j and i, with $j \in Std$, undergoes synaptic modification as follows:

$$\Delta w_{ji} = tanh(m_i/2) \cdot \eta \cdot [Ao_j o_i + Bo_j + Co_i + D]. \tag{4}$$

2.2 odNEAT: An Online Evolutionary Algorithm

odNEAT [9] is an online, distributed and decentralised version of NEAT [10]. The NEAT method, one of the most prominent neuroevolution (NE) algorithms, is capable of optimising both the topology of the network and its connection weights. NEAT starts with a population of simple networks with no hidden neurons. Topologies are gradually *complexified* by adding new neurons and connections through structural mutation. This scheme allows NEAT to find the right level of complexity for the task while avoiding *a priori* specification of the network topology. NEAT has proven successful in diverse control and decision-making problems, such as double pole balancing, outperforming several methods that use fixed topologies [11]. The important features of NEAT for the purpose of this paper are that NEAT evolves *both* the weights and the topology of an ANN, while maintaining a healthy diversity of complexifying structures simultaneously. Complete descriptions of the method are available in [9–11].

odNEAT was originally designed to run across a distributed group of agents whose objective is to evolve and adapt while operating in the environment. In this contribution, experiments were performed with a single robot. Therefore, we only describe odNEAT's important characteristics when applied to a single agent. The agent is controlled by an ANN that represents a candidate solution to the current task. The agent maintains a virtual energy level representing its task performance. The fitness value is defined as the average of the energy level, sampled at regular time intervals.

The agent maintains a set of chromosomes (the genetic encoding of candidate ANNs) and their respective fitness scores in an internal repository. The repository stores the current and previous active chromosomes. When the energy level reaches zero, the current chromosome is considered unfit for the task. A new chromosome is then created based on NEAT's genetic operators. First, two parents are selected, each one via a tournament selection of size 2. Offspring

is created through crossover of the parents' genomes and mutation of the new chromosome. Newly created chromosomes are guaranteed a minimum amount of time α during which they control the agent, a *maturation period*.

odNEAT's genetic encoding was augmented with a new modulatory neuron type in order to encode neuromodulated plasticity. Each time a new neuron is added through structural mutation, it is randomly assigned a standard or modulatory role. We augmented the genetic encoding with the learning parameters in Eq. 4. The five parameters are separately encoded and evolved in the range [-1,1] for A-D, and [-100,100] for η. It is important to note that there is no Lamarckian inheritance: weight modifications during lifetime are not passed on to offspring.

3 Experimental Setup

The concurrent foraging task used in this study is performed in an arena with different types of items that can be consumed. To assess how the robot adapts through time, we applied odNEAT with and without neuromodulation. The robot loses energy at a constant rate and must learn to find food items. There are two types of items, red items and pink items. At regular time intervals, the nutritious food items become poisonous or less nutritive and vice-versa. The robot is able to sense the type of nearby items but cannot determine the nutritive value of an item without consuming it. When an item is consumed, a new item of the same type is placed randomly in the arena. This way, the task remains dynamic while the sum of the energy value of the food items in the environment is kept constant.

The motivation for the concurrent foraging task is twofold: (i) since the robot loses energy at a constant rate, it is required to evolve efficient exploration behaviours, (ii) when a poisonous item is consumed, the robot must be able to change its food gathering policy in order to survive.

3.1 Robot Model and Behavioural Control

The simulated robot is modelled after the e-puck, a small (75 mm in diameter) differential drive robot capable of moving at speeds of up to 13 cm/s [8]. We have equipped the robot with an omni-directional camera similar to the one employed by the *s-bot* robots [16]. The image recorded is processed to calculate the distance, the red colour component, and the blue colour component of the closest object in each of the eight 45° sectors. The camera has a range of 50 cm and is subject to noise (simulated by adding a random Gaussian component within $\pm$ 5% of each of the three components' saturation value). Besides the camera, the robot has an internal energy level, comfort and discomfort sensors. The energy sensor allows the robot to perceive its virtual energy level. The remaining sensors indicate if the robot has consumed a poisonous or nutritious food item. Note that the comfort sensor does not indicate to the robot the nutritive value of a consumed food item. That information is reflected by the energy sensor, which the robot also has to learn to interpret.

The robot is controlled by an ANN synthesised by odNEAT. The ANN's connection weights $\in [-10, 10]$. The input layer consists of 27 neurons: (i) three for each 45° sector, measuring the red and blue colour components, and distance of the closest object, (ii) one neuron for each of the virtual sensors (energy, discomfort and comfort). The output layer contains three neurons, one for each wheel of the robot, and one for the gripper. The gripper enables the robot to consume the closest food item within a range of 2 cm (if any).

3.2 Experimental Parameters

The environment is a 3 x 3 meter square arena surrounded by blue walls. The virtual energy level is limited to the range [0,100] energy units. The robot is capable of surviving for approximately 17 minutes without consuming (nutritious) food as energy decreases at a rate of 0.1 units/sec. When the energy level reaches zero, a new controller is generated and assigned maximum energy (100 units). In the generation of the new controller, two parents are selected from the local repository. Crossover and mutation are performed with probabilities 0.25 and 0.4, respectively. During mutation, the probability of adding a new neuron is 0.1 while a new connection is added with probability 0.05. Each connection weight is perturbed with probability 0.02 and a maximum magnitude of 2.5. The local repository is capable of storing 30 chromosomes. Performance was found to be robust to moderate changes in these parameters.

In our experimental setup, the nutritive value of the different types of food change periodically. Periods are composed of four phases of equal duration. At the beginning of each phase, the energy value of the different types of food items is set as listed in Table 1. Each experiment lasts for 100 hours of simulated time.

Table 1. The energy value of red and pink food items during the four phases. Values listed are in energy units.

	Phase 1	Phase 2	Phase 3	Phase 4
Red item	5	8	-3	3
Pink item	3	-3	8	5

4 Results and Discussion

4.1 Effects of Neuromodulated Learning

To assess the impact of neuromodulated learning on the robot's task performance, we performed three sets of evolutionary experiments characterised by distinct phase durations p_d: (i) $p_d = 9$ min, (ii) $p_d = 90$ min, and (iii) $p_d = 900$ min. For each configuration, we placed five food items of each type and performed 30 independent runs. We consider those controllers stable that manage

to survive at least 25 times the minimum survival time, i.e., approximately 7 hours of simulated time.

The results obtained are listed in Table 2. Considering the average number of evaluations (controllers tested) required for producing stable solutions, odNEAT combined with neuromodulation required approximately 23.3% to 28.2% fewer evaluations than odNEAT without neuromodulation. odNEAT alone failed to achieve stability in two evolutionary runs, one for $p_d = 9$ min and one for $p_d = 90$ min. In these runs, the long lasting controllers operated for 4.04 and 6.69 hours, respectively. For $p_d = 9$ min and $p_d = 90$ min, differences in the number of evaluations are not statistically significant ($\rho > 0.20$ and $\rho > 0.15$ respectively, Student's t-test). For $p_d = 900$ min, the differences are statistically significant ($\rho < 0.01$). These results suggest that, as the task-requirements become more stable, so does the performance of odNEAT with neuromodulation.

Table 2. Summary of the results obtained for each of the three phase durations tested. The table lists the failure rate (runs without stable controllers), average number of evaluations required before stable solutions are evolved, and the average maximum age and gathered energy per period in each experimental setup.

Experimental setup with odNEAT				
Phase duration	Failure Rate	Evaluations	Max Age (mins)	Gathered Energy
9 mins	3.33%	39.02	3404.98 ± 1668.31	343.43 ± 35.38
90 mins	3.33%	49.28	2886.88 ± 1399.20	3491.03 ± 334.49
900 mins	0%	40.40	3041.81 ± 1446.78	42526.94 ± 6897.61
Experimental setup with neuromodulated odNEAT				
Phase duration	Failure Rate	Evaluations	Max Age (mins)	Gathered Energy
9 min	0%	29.52	3351.12 ± 1358.34	354.39 ± 46.19
90 min	0%	37.79	2799.34 ± 1650.21	3530.82 ± 336.66
900 min	0%	28.99	3074.33 ± 1283.85	45199.64 ± 6680.48

Table 3. Summary of the number of nodes and connections added to the initial network topology by each evolutionary method. Results for each configuration are averaged over 30 evolutionary runs.

Evolutionary Method	Phase Duration	Augmented Connections	Augmented Neurons
odNEAT	9 mins	26.43 ± 12.30	9.47 ± 3.95
odNEAT	90 mins	30.26 ± 17.32	10.41 ± 4.65
odNEAT	900 mins	25.17 ± 12.82	9.60 ± 4.31
odNEAT + NeuroMod	9 mins	22.89 ± 14.98	8.48 ± 4.83
odNEAT + NeuroMod	90 mins	29.32 ± 11.31	10.82 ± 3.91
odNEAT + NeuroMod	900 mins	28.91 ± 11.57	10.50 ± 3.57

Depending on the experimental setup, the most stable controller of each run operated from approximately 47 hours to 57 hours of simulated time before the experiment was terminated. This result indicates that the evolutionary process is capable of evolving controllers well adapted to the periodic changes in the nutritive value of the food items. In terms of gathered energy per period, neuromodulated solutions perform slightly better. ANNs evolved with and without neuromodulation have a similar topological complexity. The initial topology of stable solutions was augmented with a comparable number of connections and neurons (see Table 3). Topologies are synthesised faster by odNEAT with neuromodulation. This result suggests that when neuromodulation is present, odNEAT performs a more efficient exploitation of a given network topology. In fixed-weight networks, fine-grained adjustment of connection weights can only be achieved through mutation. Modulated networks allow for a different expression of a given topology's potential, and are advantageous even when task requirements do not change for a long time (p_d= 900 mins). When modulatory neurons are present, solutions are synthesised after fewer controller evaluations, probably due to the modification of internal dynamics by each network.

4.2 Structural Role of Neuromodulation

The results presented above show that neuromodulated learning allows for faster synthesis of stable controllers. In this section, we analyse the structural role of neuromodulation on the most stable controllers of each independent run in order to determine how it affects internal neural dynamics.

Table 4. Summary of the most stable controllers in each independent run. The table lists the augmented and modulatory neurons, and augmented and modulatory connections in each network.

Phase Duration	Aug. Neurons	Mod. Neurons	Aug. Connections	Mod. Connections
9 mins	9.73 ± 4.88	4.97 ± 2.92	23.93 ± 13.28	6.37 ± 4.39
90 mins	11.97 ± 4.02	6.07 ± 2.99	30.57 ± 10.58	7.93 ± 4.34
900 mins	10.10 ± 5.07	5.03 ± 3.36	25.67 ± 13.34	6.97 ± 4.90

Table 4 shows the average complexity of each stable solution. Approximately half of the augmented neurons have a modulatory role. Modulatory actions are localised as each of these neurons typically connects to one or two other neurons. A common topological characteristic between evolved solutions is that the majority of modulatory connections have output neurons as targets. In fact, the evolutionary process often leads to the appearance of specialised neurons that exclusively regulate output neurons as listed in Table 5. Depending on the experimental setup, 59% to 69% of the modulatory neurons are specialised units. 6% to 9% of the specialised neurons modulate more than one output neuron. For $p_d = 9$ and $p_d = 900$ mins, differences in the number of specialised neurons are statistically significant ($\rho < 0.05$, Student's t-test). Analysis of experimental

data shows that there is a higher regulatory activity of outputs for the setups of $p_d = 9$ mins and $p_d = 90$ mins. In these scenarios, controllers experience more environmental changes during task-execution. Food gathering policies must be flexible and change whenever a nutritious item becomes less nutritive or poisonous. With the increase of phase durations, the task becomes less dynamic and the percentage of specialised neurons decreases. Existing specialised neurons increasingly focuses on movement (left and right wheels) and less on the gripping and food consumption actions.

Table 5. Summary of the specialised neurons for the best solutions of each evolutionary run. The table lists the percentage of modulatory neurons that are specialised in regulating the output actions, and the percentage of specialised neurons that regulate each output. LW and RW represent the left and right wheel, respectively.

Phase Duration	Specialised Reg. Neurons (%)	LW (%)	RW (%)	Gripper (%)
9 mins	69 ± 20	34	34	38
90 mins	62 ± 24	40	32	35
900 mins	57 ± 26	50	30	29

5 Conclusions and Future Work

In this paper, we have introduced a novel approach to the online synthesis of behavioural control for autonomous robots. We combined odNEAT and neuromodulated learning. While odNEAT evolves online both the weights and the topology of neural controllers, neuromodulation allows each individual controller to actively modify its internal dynamics. We demonstrated our method through a series of simulation-based experiments in which an e-puck-like robot had to perform a dynamic concurrent foraging task. We showed that odNEAT with neuromodulation outperforms simple odNEAT by requiring fewer evaluations to produce stable solutions. Results indicate that neuromodulated learning is beneficial even when task requirements do not change for a long time.

We showed that the evolutionary process generates controllers well adapted to the periodic changes in the nutritive value of the food items. Depending on the experimental setup, the most stable controller in each run operated from approximately 47 hours to 57 hours of simulated time before the experiment was terminated. The controllers had thus become resilient to changes in task requirements and they could have operated for longer if they had been given more time. In order to determine the structural and functional role of neuromodulation, we analysed the evolved topologies of the most stable solutions. Evolved networks are characterised by specialised neurons dedicated to regulating the output neurons.

The immediate follow-up work to this study includes the analysis of the neural activation patterns and weight changes to better understand the neural dynamics and the decision-making mechanisms underlying the robot's behaviour.

References

1. Floreano, D., Keller, L: Evolution of Adaptive Behaviour in Robots by means of Darwinian Selection. PLoS Biology 8, 1–8 (2010)
2. Watson, R.A., Ficici, S.G., Pollack, J.B.: Embodied evolution: Embodying an Evolutionary Algorithm in a Population of Robots. In: 1999 Congress on Evolutionary Computation, pp. 335–342. IEEE Press, Piscataway, NJ (1999)
3. Huijsman, R.J., Haasdijk, E., Eiben, A.E.: An on-line on-board distributed algorithm for evolutionary robotics. In: Hao, J., Legrand, P., Collet, P., Monmarché, N., Lutton, E., Schoenauer, M. (eds.) 10th International Conference on Evolution Artificielle (EA 2011), pp. 119–129. [Online Proceedings] - `http://www.info.univ-angers.fr/ea2011/doc/EA2011_ProceedingsWeb.pdf`, (2011)
4. Haasdijk, E., Eiben, A. E., Karafotias, G.: On-line Evolution of Robot Controllers by an Encapsulated Evolution Strategy. In: 2010 IEEE Congress on Evolutionary Computation, pp. 1–7. IEEE Press, Piscataway, NJ (2010)
5. Hinton, G.E., Nowlan, S. J.: How Learning can Guide Evolution. Complex Syst. 1(3), 495–502 (1987)
6. Floreano, D., Urzelai, J.: Evolutionary Robots with On-line Self-organization and Behavioral Fitness. Neural Netw. 13, 431–443 (2000)
7. Urzelai, J., Floreano, D.: Evolution of Adaptive Synapses: Robots with Fast Adaptive Behavior in New Environments. Evol. Comput. 9, 495–524 (2001)
8. Mondada, F., Bonani, M., Raemy, X., Pugh, J., Cianci, C., Klaptocz, A., Magnenat, S., Zufferey, J., Floreano, D., and Martinoli, A.: The e-puck, a Robot Designed for Education in Engineering. In: 9th Conference on Autonomous Robot Systems and Competitions, vol. 1, pp. 59–65. IPCB, Castelo Branco, Portugal (2009)
9. Silva, F., Urbano, P., Oliveira, S., Christensen, A. L.: odNEAT: An Algorithm for Distributed Online, Onboard Evolution of Robot Behaviours. In: 13th International Conference on the Simulation and Synthesis of Living Systems (ALIFE13), pp. 251–258. MIT Press, Cambridge, MA (2012)
10. Stanley, K. O., Miikkulainen, R.: Evolving Neural Networks through Augmenting Topologies. Evol. Comput. 10(2), 99–127 (2002)
11. Stanley, K. O.: Efficient Evolution of Neural Networks through Complexification. PhD thesis, the University of Texas at Austin, Austin, TX (2004)
12. Soltoggio, A., Bullinaria, J. A., Mattiussi, C., Dürr, P., Floreano, D.: Evolutionary Advantages of Neuromodulation Plasticity in Dynamic, Reward-based Scenarios. In: 11th International Conference on the Simulation and Synthesis of Living Systems (ALIFE11), pp. 569–579. MIT Press, Cambridge, MA (2008)
13. Niv, Y., Joel, D., Meilijson, I., Ruppin, E.: Evolution of Reinforcement Learning in Uncertain Environments: A Simple Explanation for Complex Foraging Behaviors. Adapt. Behav. 10(1), 5–24 (2002)
14. Katz, P.: Beyond Neurotransmission: Neuromodulation and Its Importance for Information Processing. Oxford University Press, Oxford (1999)
15. Bailey, C. H., Giustetto, M., Huang, Y.-Y., Hawkins, R. D., Kandel, E. R.: Is Heterosynaptic Modulation Essential for Stabilizing Hebbian Plasticity and Memory?. Nat. Rev. Neurosci. 1(1), 11–20 (2000)
16. Ampatzis, C., Tuci, E., Trianni, V., Christensen, A. L., Dorigo, M.: Evolution of Autonomous Self-Assembly in Homogeneous Robots. Artif. Life 4(15), 465–484 (2009)