The Neuroevolution of Teaching Architectures in an Artificial Life Model

Edward Robinson

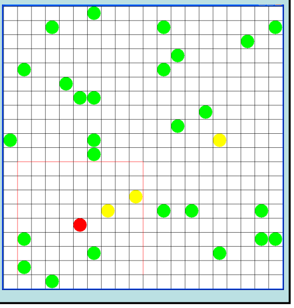
School of Computer Science, University of Birmingham

Introduction

A promising area of research is that which mixes evolutionary and learning techniques—particularly as it has been shown that learning can enhance the adaptive power of evolution. In artificial life studies animats are often controlled by fixed neural network architectures. Several learning techniques have been combined with the neuroevolution of such controllers, including, supervised learning, reinforcement learning and hebbian learning. Here we examine a body of research which uses ‘auto-teaching networks’ that generate their own teaching input and provide an error by which the network controller can be trained. Further, we situate animats in a dynamic environment—more complex than previously used in other work.

A Dynamic Environment

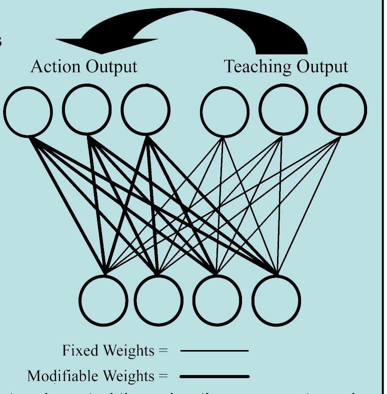
Animats live in a 2d grid-world populated with plants (green) that randomly sprout and dwindle with different frequency. An animat relies on the energy obtained from the plants to survive. However, being greedy and consuming all of the plants is not such a good idea since the crop is seasonal and will not sprout during winter months. In order to survive a full year or more an animat must bury some of the plants (yellow) so it can eat during the winter.



Network Architecture

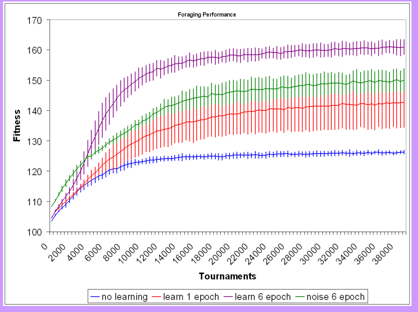
Each animat is controlled by a feed-forward neural network. The networks have two sets of outputs connected to the sensory inputs. The teaching outputs are connected via fixed weights. The output values are used to adjust the modifiable weights connected to the action outputs via back-propagation during the animat’s lifetime. Both the teaching weights and the birth action weights are inherited.

Sensory Inputs: Animats receive the relative position of the nearest plant, the current time of season and their energy level.

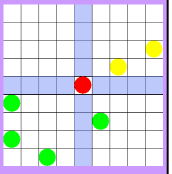


Action Outputs: The outputs of the network control the animat’s movements and whether it consumes or buries a plant.

Foraging Behaviour

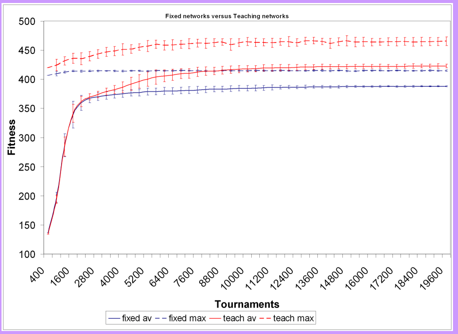


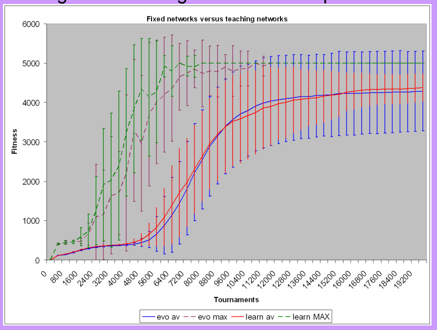
Here we examined the possibility that teaching networks could improve foraging performance when the animats were not given the position of the nearest plant and instead had to explore the environment. Each animat had sensor range (right) in which plants could be detected. Again we see that animats with teaching networks outperform fixed networks, and animats with noise based architectures—suggesting that the plasticity is not generating random noisy policies but directed evolved ones.

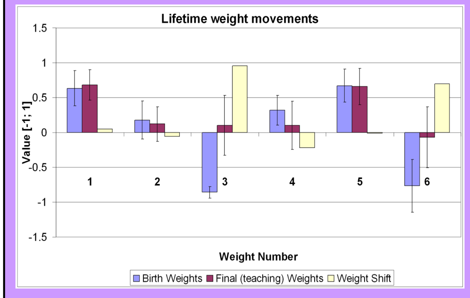


Teaching Nets versus Fixed Nets

Figure 3: Animat performance between teaching and fixed nets (below right). Performance on same task when animats are given less sensory information (below left). Analysis of how the weights shift during the lifetime of a plastic animat (bottom left)







In the first experiment we evolved animats in the environment shown on the left of the poster and with the architecture above. We wanted to see if teaching networks improve animat performance[1]. In contradiction to [1] we can see there is no significant difference between animats that are fixed or plastic, although the plastic animats results have less variability (top fig).

Next we removed sensory information from the animats and configured their networks so they were only given the position of the nearest plant but no other internal ‘proprioceptors’. Results now indicate a significant increase in performance of animats that had lifetime learning (best results were significant with a 99.9% confidence level).

We can see from the weight movement graph that evolution selects for different policies in the two networks and that it’s the interaction between the birth network and the teaching network that is responsible for the improved performance. At birth the animat chooses to bury any plants it finds for the winter (w 3 + w 6), but as the learning takes place these weights are significantly shifted so that animats begin to consume plants they find. This means that when winter arrives animats have stores ready to consume.

Discussion

We have shown that auto-teaching networks do not always out-perform fixed networks on certain tasks. Learning has costs (such as learning the wrong thing) and if the benefits gained from plasticity do not outweigh these costs then rigid architectures will outperform plastic ones. However, we have shown that in some cases, where it is difficult to find a policy given the available weight-space, auto-teaching networks can provide a level of adaptability that can improve performance. Not by specifying a better policy alone, but by adapting lifetime behaviour in such a way that generates a policy that a rigid network could not create alone. Finally, we have seen from the exploratory foraging experiment that the policies generated by the interactions between fixed and teaching networks are better than policies that can be created using fixed networks are noise injection techniques.

References

[1] Nolfi S., Parisi D. (1993). Auto-teaching: networks that develop their own teaching input, In: J.L. Deneubourg, H. Bersini, S. Goss, G. Nicolis, R. Dagonnier (Eds.) Proceedings of the Second European Conference on Artificial Life, Brussels.