The Evolution of General Intelligence

Padmini Rajagopalan[1], Kay E. Holekamp[2] and Risto Miikkulainen[1]

[1]University of Texas at Austin, Austin, TX 78712
[2]Michigan State University, East Lansing, MI 48824
padmini@cs.utexas.edu

Abstract

When studying different species in the wild, field biologists can see enormous variation in their behaviors and learning abilities. For example, spotted hyenas and baboons share the same habitat and have similar levels of complexity in their social interactions, but differ widely in how specific vs. general their behaviors are. This paper analyzes two potential factors that lead to this difference: the density of connections in the brain, and the number of generations in prolonged evolution (i.e. after a solution has been found). Using neuroevolution with the NEAT algorithm, network structures with different connectivities were evaluated in recognizing digits and their mirror images. These experiments show that general intelligence, i.e. recognition of previously unseen examples, increases with increase in connectivity, up to a point. General intelligence also increases with the number of generations in prolonged evolution, even when performance no longer improves in the known examples. This outcome suggests that general intelligence depends on specific anatomical and environmental factors. The results from this paper can be used to gain insight into differences in animal behaviors, as well as a guideline for constructing complex general behaviors in artificial agents such as video game bots and physical robots.

Introduction

All species have special abilities that help them survive in their ecological and social niche. Such abilities can be termed *domain-specific intelligence*. For example, spotted hyenas are extremely capable hunters, whether alone or in a pack, but they are poor at tasks in new domains to which they have not been exposed. On the other hand, baboons, which share the same habitat and have similar social structure, are very good at solving new tasks that they have never seen before, such as finding food that is hidden from them by human beings. This ability to solve novel problems outside of specific cognitive domains is called *domain-general intelligence*. Thus, the extent of general intelligence is different in different species, even those in similar environments.

A prevailing theory is that domain-general intelligence emerges from the interaction of multiple processes or modules in the brain (Kaufman et al., 2011; van der Maas et al., 2006). So far, there have been few attempts to analyze or verify this theory through computer simulations. In this paper, neuroevolution techniques were used to study the evolution of general intelligence. These simulations investigated the effects of different network structures and prolonged evolution on the ability to solve a new task never seen during evolution. Network connection densities were varied to resemble the interconnectivity of brain modules in different species. The length of prolonged evolution after the old tasks had been solved was also varied. It was discovered that the performance on the new task, i.e. general intelligence, increased with increasing connection density, up to a point. Performance also improved with prolonged evolution even though fitness in the old tasks no longer improved. These results suggest that general intelligence may emerge due to specific anatomical and environmental factors.

Neural networks were employed in these experiments because they have been used to simulate brain structures previously with success(Amit, 1992; Miikkulainen et al., 2005). In addition, neuroevolution has been used to study predator-prey interactions and the evolution of complex behaviors in animals (Rawal et al., 2010; Rajagopalan et al., 2011). Thus the insights from this paper should be useful in building better better models of intelligent behavior in the wild, as well as to develop more realistic artificial agents in video games or robotics.

This paper is organized as follows. The next section examines the biological background and prior work in analyzing the origins of general intelligence in animals. The neuroevolution architecture and algorithm used in this paper are discussed in the section after that. The two sections following that lay out the hypotheses and the experimental setup used to test them. The Results section describes the actual experiments performed as well as the results obtained.

Biological Background

Animals in their natural habitat face many different kinds of problems for which special abilities may be required. Indeed, all species exhibit specialized cognitive skills that are essential for their survival. These skills fall under the realm of domain-specific intelligence, and may be activated or in-

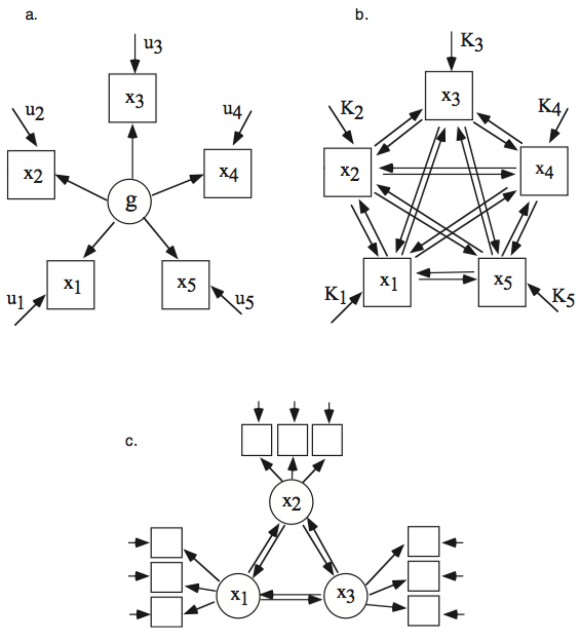

Figure 1: Three models of the positive manifold: (a) the standard *g* model, which shows the underlying *g* factor that influences all cognitive processes, (b) the mutualism model, showing the interaction between various mental mechanisms, and (c) the extended mutualism model, where only the latent processes interact. Squares and circles denote manifest and latent variables, respectively. Symbols x denote processes, u unique variances, and K resources (van der Maas et al., 2006).

hibited by specific circumstances arising in the environment. Each ability is thought to be controlled by a particular module or area of the brain (Cosmides and Tooby, 2002). Thus, many animal species excel in solving certain problems, but not others.

In contrast to domain-specific intelligence, domain-general intelligence is the ability of an individual to solve new problems. Some animal species, for example, mice (Matzel et al., 2011) and primates (Reader and Laland, 2002), possess an underlying cognitive mechanism that helps them with associative learning and general problem-solving. Evolutionary biologists and behavioral ecologists have posited that domain-general intelligence consists of several linked domain-specific abilities (Cosmides and Tooby, 2002). It has also been hypothesized that general intelligence emerged from domain-specific capabilities evolving under complex evolutionary pressures. Primatologists argue that these pressures were specifically the complex social structures and labile social interactions required of primates (Dunbar, 2003; Byrne and Whiten, 1989).

Primates are well known to be superior to most other animals in their behavioral flexibility and ability to solve problems (Byrne and Whiten, 1989; Tomasello and Call, 1997). Moreover, they live in structurally complex groups. Therefore, social cognition seems to be related to general intelligence, implying that all animal species that live in similarly large and complex societies should possess primate-like cognitive abilities. However, this prediction is not always true. For example, the societies of spotted hyenas (*Crocuta crocuta*) are remarkably like those of cercopithecine primates such as savanna baboons (*Papio anubis*) with respect to size, composition, structure, and patterns of competition and cooperation (Holekamp et al., 2012). A host of studies have found that spotted hyenas and baboons have converged with respect to social cognition (Holekamp et al., 2007; Benson-Amram et al., 2011), but there is no evidence that their domain-general intelligence has converged. This result indicates that there may be some fundamental differences in the way the brain structures and cognitive faculties of different species have evolved.

General intelligence is most apparent in human beings. There is a significant positive correlation between abilities in various cognitive tasks (van der Maas et al., 2006): A good score on one cognitive test predicts good scores on all other cognitive tests. This empirical phenomenon is called the *positive manifold*, and it is unlikely to result from a strictly modular brain (Figure 1).

There are many theories regarding the origin of the positive manifold (van der Maas et al., 2006). One hypothesis is that there must be a single underlying mechanism in the brain on which general intelligence depends (van der Maas et al., 2006; Sternberg and Grigorenko, 2002). This factor is commonly denoted as *g* (Figure 1a). Another general intelligence model is called *mutualism* (van der Maas et al., 2006). In this model, the positive manifold arises from the interaction of multiple cognitive processes in the brain (Figure 1b). The extended mutualism model (Figure 1c) posits that *g* itself arises from the interaction between several latent cognitive mechanisms in the brain. These latent variables then influence the outwardly manifest variables that constitute general intelligence (van der Maas et al., 2006).

In this paper, neuroevolution was used to evaluate the theory of mutualism in the evolution of general intelligence. Neural networks represented the brains, and the neurons in them were the cognitive mechanisms that interacted to produce general intelligence. The extent of interactions was characterized by the number of connections between neurons in a neural network. Hence, a neural network with more connections was expected to perform better when tested on a general intelligence task. In addition, the amount of task-specific learning was varied to check whether there was a positive manifold or correlation between performance in that task and performance in a new cognitive task.

NeuroEvolution of Augmenting Topologies

Neuroevolution has previously been used to great effect for producing dynamic and intelligent behaviors in autonomous agents. For example, it has been used in simulated robot soccer (Whiteson et al., 2005), robotic battle (Stanley and Miikkulainen, 2004) and Ms. Pac-Man (Burrow and Lucas, 2009). It has also been used to simulate the hunting and social behaviors of animal groups (Rawal et al., 2010; Rajagopalan et al., 2011; Rawal et al., 2012). Thus, neuroevolution is a natural choice for modeling the emergence of general intelligence.

NeuroEvolution of Augmenting Topologies, or NEAT (Stanley and Miikkulainen, 2002), is a neuroevolution technique that optimizes not only the connection weights, but also the topology of a neural network. NEAT has the resources to add and delete both nodes and links. This technique was shown to be more effective than traditional neuroevolution methods that modify only the connection weights of neural networks (Stanley and Miikkulainen, 2002).

To make crossover between two neural networks with different topologies possible, their links have to be lined up according to the nodes they connect. NEAT makes this possible using a historical marker called a global innovation number which records when each connection was formed in the population's history. Speciation is also used to nurture new innovations in network structure that might otherwise be lost due to their low initial fitnesses.

In this paper, the NEAT algorithm was used to evolve simple neural networks. These networks were trained on one cognitive task, but tested on a different task to study their ability to adapt. The number of links in the neural networks was varied to analyze the effect of connection density on performance in general intelligence, or the testing phase. It is easy to manipulate the connectivity of networks in NEAT by changing the probability of adding new links. This ability makes it a good choice as the neuroevolution algorithm for this particular study.

Hypotheses

The experiments in this paper were conducted with the objective of simulating the early evolutionary processes in the brains of species with different levels of domain-general intelligence. With this goal in mind, two factors thought to influence general intelligence were tested. The following hypotheses specify the expected results:

1. *The general intelligence should increase with the connection density.*

   Theories from differential psychology claim that more interaction between the various modules in the brain that have evolved for specific tasks results in higher domain-general intelligence (van der Maas et al., 2006). In the simulations in this paper, a neural network with more connections between its neurons is expected to have perform better in the testing phase.

2. *The general intelligence should increase with extended evolution.*

   As stated in the Biological Background section, performance in various cognitive tasks are found to be positively correlated with one another. If neural networks are evolved past the point where their performance in the known tasks has plateaued, they evolve to be more diverse, robust and efficient in these tasks (Watson et al., 2011; Lehman and Stanley, 2010). They are more resilient to changes in network function (such as mutations), and should be more resilient to changes in the input as well, and thereby more general. As a result, their performance in the testing phase should improve with prolonged evolution on the first task.

Experimental Setup

This section describes the experimental setup used for the experiments in this paper. As explained in the previous sections, the goal was to computationally explore theories from neurobiology and psychology about the evolution of domain-general intelligence. To simulate the evolution of animal brains, simple two-layer neural networks were evolved using the NEAT algorithm.

The connections between the neurons simulated the interaction of processes in the brain, and the length of prolonged evolution (i.e. evolution after a satisfactory performance had been reached) represented the opportunity to evolve generalized processes. In the testing phase, new input was presented to the neural network. These inputs were related to the old inputs, but were not shown to the neural network during evolution. The performance on the new input was taken as the general intelligence of that particular neural network. Fitness was defined as the percentage of inputs for which the correct output was produced.

All neural networks in the population were identical before the start of evolution. They each had a single hidden layer of neurons and sigmoidal activation functions. In the preliminary experiments, topology as well as weights could change through evolution. The later experiments involved the evolution of only the connection weights. Even when topology was evolved, only links were added or deleted; the number of hidden neurons was always constant. The population consisted of a single species of either 30 or 100 individual neural networks. This number and the number of species changed during the program's execution as part of the NEAT algorithm. Each experiment was run 20 times for each population of neural networks and their results were averaged.

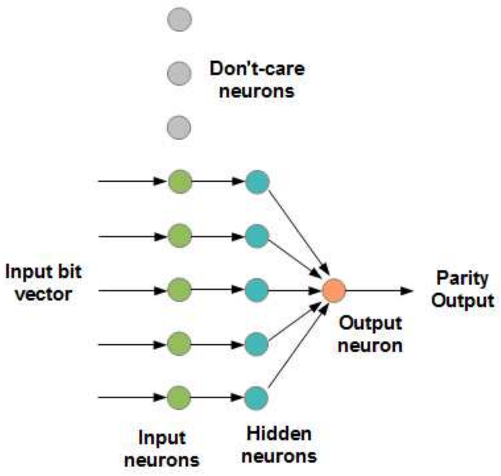

Figure 2: The initial network topology for the parity task. There are eight input nodes of which five are active, and three are don't cares. The five input nodes are connected to five hidden neurons, each of which is connected to the single output node. This preliminary task demonstrated that the topology-evolving approach would converge to similar architectures for this simple problem. Therefore, a fixed-topology approach in a more challenging task was necessary to test the hypotheses.

Results

Preliminary experiments were first conducted in the multi-bit parity domain to check what kinds of tasks and neural network architectures were most suitable for simulating general intelligence. In these experiments, both the connection weights and the structure of the networks were evolved using NEAT. Although links between already existing neurons were added or removed during evolution, neurons themselves were not created or deleted. Thus, the only factors affecting fitness were the number and location of the links, and the values of their weights. In this manner, interconnections between various brain modules were modeled. Insights from these experiments were then used to design a second set of experiments on a handwritten digit recognition domain.

Parity Task for Topology-Evolving Networks

For practical reasons, it is useful to start any simulation study with the simplest of tasks and the most basic neural network structures. After gaining insights on how to make them work, the algorithm and the task may be expanded gradually to more complex tasks. In this case, a simple logic function, multi-bit parity, served as a starting point to test the neural networks. In this task, the answer is 1 if the number of 1's in the input is odd, and 0 if the number of 1's in the input is even. The neural networks had eight neurons or nodes in the input layer, five nodes in the hidden layer and a single output neuron (Figure 2). Initially, each of the five hidden neurons was connected to a different input neuron. Due to the sigmoidal activation function, the output neuron produced a real-valued number between 0 and 1. This value was taken to mean a 0 if less than 0.5, and a 1 otherwise. Of the eight input nodes, five were in use and had incoming multi-bit input, one bit at each node. Hence, there were 32 possible input bit strings. The other three nodes had as input fixed "don't-care" values. The neural networks were expected to ignore the don't-care inputs while successfully calculating the parity of the correct inputs.

Several combinations of evolution and testing scenarios were given to the networks and the percentage of testing phase inputs for which the network gave the wrong output (error) was recorded. Several neural network populations with different probabilities of adding links during evolution were tested in each of these. The idea was that networks with a higher probability of having links added would have acquired more links by the end of the simulation. Thus, according to our original hypotheses, these networks should perform better in the testing phase.

More specifically, the following experiments were performed in the parity task:

1. *Evolution task*: 22 of the 32 possible input bitstrings; don't care = 0

   Test set: The other 10 bitstrings; don't care = 0

2. *Evolution task*: 22 of the 32 possible input bitstrings; don't care = 0

   Test set: The other 10 bitstrings; don't care changes to 1

3. *Evolution task*: All 32 bit combinations; don't care = 0

   Test set: All 32 bit combinations; don't care = 1

The purpose of conducting these different experiments was to evolve a neural network on one set of inputs and then test it on another, related set. If the neural network had evolved to generalize, the test problem should be solvable even if never seen before by the network. The performance on the new inputs was also expected to depend on the number of connections between the neurons of the network in question.

The main result was that all the different populations of neural networks (with the different probabilities of adding links during evolution) had similar performances during the testing phase. The reason was that all the networks had similar structure at the end of the simulation. Apparently, the NEAT algorithm is capable of adding links as needed for different network structures regardless of the probability of adding links. Moreover, although some links had been added for every network, the final structures still remained very simple. This result proved that the parity task was solvable by too simple a network structure. Therefore, a more

```
0 1 1 1 1 1 1 0
0 0 0 0 0 0 1 0
0 0 0 0 0 1 1 0
0 0 0 0 0 1 0 0
0 0 0 0 1 1 0 0
0 0 0 1 1 0 0 0
0 0 1 1 0 0 0 0
0 0 1 0 0 0 0 0
```

Figure 3: An example of the 8×8 bitmap encoding the image of a single digit, 7. The 1's represent black and the 0's represent white. This bitmap is concatenated into a 64-bit vector, which is given as input to the neural networks.

complex problem as well as a different neuroevolution strategy was needed to test the hypothesis that more connections mean more general intelligence.

Handwritten Digit Recognition Task for Fixed-Topology Networks

To avoid the problem with the parity experiments, in the second set of experiments, the topologies of all the networks were fixed and only the connection weights were evolved. Twenty trials of each experiment were run for each population and their results were averaged. The connection density remained the same between trials, but the individual connections could be different. Both feed-forward and recurrent neural networks were tested.

The recognition of handwritten digits, a well-known benchmark problem for neural networks, was chosen as a more challenging task, adapted for a test for domain-general intelligence. The inputs were not only the images of the digits, but also their mirror images. The mirror image of a single digit, 7, was left out of the set of inputs used during evolution and only used as a test input. The digit 7 was chosen for this purpose because the mirror image of 7 is not easily confused with other digits, and 7 is distinct from its mirror image (unlike 0, 1 or 8). If the neural network had evolved to recognize that some of its inputs were mirror images of other inputs, it should have no difficulty in recognizing a previously unseen set of mirror images. The performance during the testing phase is thus a good measure of the extent to which general intelligence has evolved in a neural network.

The handwritten digits were obtained from data stored in the NIST database. These images are taken from real-world data and therefore contain several noisy and distorted examples. The image of each digit was converted into an 8×8 bitmap representation, where 0 represents white and 1 represents black (Figure 3). The bits in the bitmap were then concatenated into a 64-bit vector, which was used as a single input to the neural network. Therefore, the neural networks in these experiments comprised 64 input nodes and 10 output nodes, representing a digit from 0 to 9. The digit corresponding to the node with the maximum value of output was considered the answer of the neural network to that particular input. The number of hidden neurons in each neural network was fixed at 25 (Figure 4). The number of distinct handwritten digit images shown to the networks during evolution was 5687, and the number of mirror images of 7 shown during the testing phase was 299.

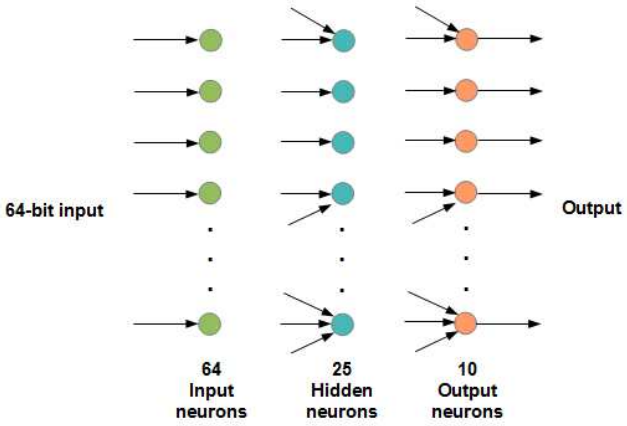

Figure 4: The network topology for the handwritten digit recognition task. There are 64 input nodes, 25 hidden neurons and ten output nodes. The connections between them are initially added at random based on the required connection density, but remain fixed thereafter. This more complex task and fixed-density architecture allowed testing the hypotheses about connectivity and prolonged evolution reliably.

The fixed topology of the neural networks meant that the effect of connection density on the testing phase performance could be easily measured. The second hypothesis, that the extent of prolonged evolution influences test performance, could also be verified by varying the number of generations for which the neural network was evolved after it had already evolved to recognize the inputs shown during evolution. The results of these experiments are shown in figures 5 and 6.

As predicted, the test performance increased with increasing connection density of the neural network. This result was true for both feed-forward and recurrent neural networks. After 65% density, however, the test performance started to decrease again (Figure 5). At the same time, the fitness on the training inputs also decreased, indicating

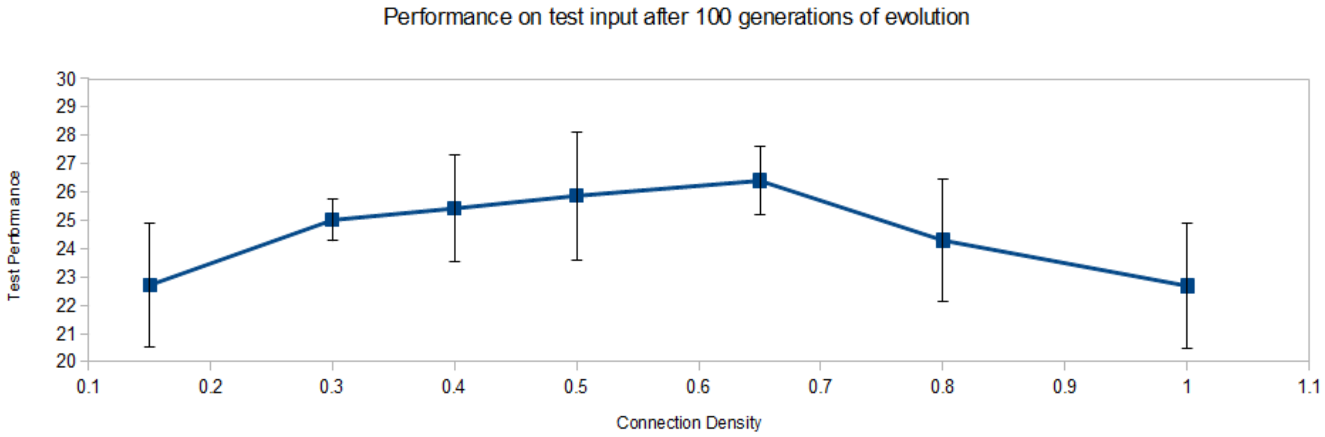

Figure 5: The average performance on the test phase (with unseen input) after evolving the network for 100 generations on the other inputs. Connection density fraction is on the *x*-axis and percentage of test input images the neural network identified correctly on the *y*-axis. The performance increases with increased connectivity up to a point, indicating more general intelligence.

that the decrease was not due to overfitting, but probably from the difficulty of evolving large numbers of connection weights.

Similarly, the test performance also increased with increasing numbers of generations for which the neural network was evolved. Evolution was continued beyond the 100 or so generations needed for networks with the optimal connection density (65%) to learn the training inputs. The performance on unseen inputs was plotted against number of generations for which the networks were evolved on the training inputs (Figure 6). It should be noted that there was no significant increase in the fitness over the evolution examples during the prolonged training. There was no effect on the results whether population size was 30 or 100. The fitness in Figure 6 for the point 0 on the *x*-axis (100 generations of evolution, stagnation just reached) is the same as the fitness for point 0.65 in Figure 5.

Thus, these results confirm that both denser connectivity and prolonged evolution result in networks that implemented more general solutions and therefore, more general intelligence.

Discussion and Future Work

The experiments described in the previous section support the hypothesis that fitness on a general intelligence task increases with increased connectivity in the brain. With the particular tasks with the simple neural networks in these experiments, more connections typically equals better generalization ability.

After a certain point, the testing performance decreases again. This is not due to overfitting but a different mechanism. The likely reason is that the neural network becomes too complex for evolution to optimize. The networks have 64 input neurons, 25 hidden neurons and 10 output neurons each; at 0.65 connection density (i.e. with 65% of all possible connections), the number of connection weights that need to be evolved becomes very large. Excessive complexity would also explain why the populations of recurrent neural networks generally have a lower test performance than populations of feed-forward neural networks. Not only is the number of connections large, but the number of timesteps required for the network to settle is also higher.

From machine learning perspective, more connections in a neural network and prolonged training should result in overfitting, i.e. the network should not be able to generalize as well. It is interesting that the opposite effect is observed in the experiments in this paper: more connections and prolonged evolution results in more general behavior. Note that the task in this paper is not to simply interpolate between training examples, which is usually done in machine learning experiments, but to extend the behavior to a new class of examples. Such a task is more cognitive, and indeed the result is different. The task is still limited and it is difficult to demonstrate extensive general intelligence in it. But the fundamental property of general intelligence is the ability use behaviors in a novel context. Thus, it can be argued that the results indicate not just a generalization of neural networks, but the signature of general intelligence.

One significant insight gained from the above simulations is that the neural network architecture, the training algorithm, and the cognitive task employed to test general intelligence are all important. Further work is necessary to determine which combinations are most conducive to general intelligence. The main advantage of NEAT is the evo-

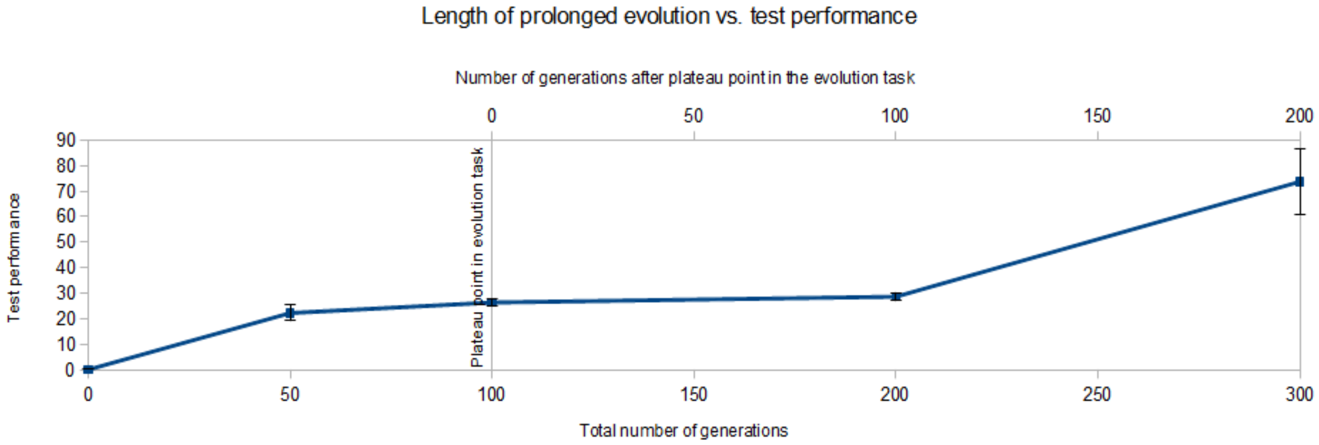

Figure 6: The average test performance of neural networks with connection density 65% after evolution for 50, 100, 200, and 300 generations on the other inputs. The training fitness usually plateaued at around 100 generations of evolution. The total number of generations is on the *x*-axis below the plot and the number of generations after fitness stagnation on the *x*-axis above the plot. The percentage of test input images the neural network identified correctly is on the *y*-axis. The performance on the test input increases with prolonged evolution.

lution of neural network structure, but since it is the level of interaction between many specialized modules that is being studied, a fixed-structure neuroevolution algorithm such as Multi-Component ESP (Gomez and Miikkulainen, 1997; Rawal et al., 2010) may be more beneficial. Different subnetworks of the same network can be evolved to perform different tasks, and their interconnections can be varied while the whole network is tested on a new domain-general task. Preliminary experiments with a hybrid NEAT-ESP algorithm are quite promising.

Similarly, the tasks themselves can be modified to resemble those faced by real animals in the wild better. Neuroevolution has already been used to evolve complex animal behaviors such as group hunting, communication with conspecifics and evading predators (Rawal et al., 2010; Rajagopalan et al., 2011; Rawal et al., 2012). Therefore, a task extension in this direction should be possible.

General intelligence approaches may ultimately be used to design robots or video game characters with realistic and complex behaviors. Such agents should be more versatile, more engaging, and more effective than is currently possible.

Conclusions

In this paper, two factors believed to influence the evolution of domain-general intelligence were tested using neuroevolution: connection density and prolonged evolution. The results indicate that increasing the number of connections leads to an increase in the general intelligence, up to a point. They also verify the hypothesis that prolonged evolution in similar cognitive tasks leads to better performance in a previously unseen task. Hence, it can be concluded that general intelligence is determined by specific anatomical and environmental factors that affect the evolution of an animal species.

The simulations from this paper can be extended in the future to more realistic tasks in environments resembling those in which real-life animal species live, gaining insight into differences observed in biology. Eventually, the same approach could be useful in creating intelligent behaviors for artificial agents in video games or robotics as well.

Acknowledgements

This research was supported in part by NSF grants DBI-0939454 and IIS-0915038, and in part by NIH grant R01-GM105042.

References

Amit, D. J. (1992). *Modeling brain function: The world of attractor neural networks*. Cambridge University Press.

Benson-Amram, S., Heinen, V. K., Dryer, S. L., and Holekamp, K. E. (2011). Numerical assessment and individual call discrimination by wild spotted hyaenas, (*Crocuta crocuta*). *Anim. Behav.*, 82(4):743–752.

Burrow, P. and Lucas, S. M. (2009). Evolution versus temporal difference learning for learning to play Ms. Pac-man. In *Proceedings of the IEEE Symposium on Computational Intelligence and Games (CIG 2009)*, pages 53–60.

Byrne, R. and Whiten, A. (1989). *Machiavellian intelligence: social expertise and the evolution of intellect in monkeys, apes, and humans*. Oxford Science Publications.

Cosmides, L. and Tooby, J. (2002). Unraveling the enigma of human intelligence: Evolutionary psychology and the multimodular mind. *The evolution of intelligence*, pages 145–198.

Dunbar, R. I. M. (2003). The social brain: mind, language, and society in evolutionary perspective. *Ann. Rev. Anthropol.*, 32:163–181.

Gomez, F. and Miikkulainen, R. (1997). Incremental evolution of complex general behavior. *Adaptive Behavior*, 5(3-4):317–342.

Holekamp, K. E., Sakai, S. T., and Lundrigan, B. L. (2007). Social intelligence in the spotted hyena (*Crocuta crocuta*). *Phil. Trans. Roy. Soc. B.*, 362(1480):523–538.

Holekamp, K. E., Smith, J. E., Strelioff, C. C., Horn, R. C. V., and Watts, H. E. (2012). Society, demography and genetic structure in the spotted hyena. *Molec. Ecol.*, 21(3):613–632.

Kaufman, S. B., DeYoung, C. G., Reis, D. L., and Gray, J. R. (2011). General intelligence predicts reasoning ability even for evolutionarily familiar content. *Intelligence*, 39(5):311–322.

Lehman, J. and Stanley, K. O. (2010). Revising the evolutionary computation abstraction: minimal criteria novelty search. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, pages 103–110. ACM.

Matzel, L. D., Wass, C., and Kolata, S. (2011). Individual differences in animal intelligence: learning, reasoning, selective attention and inter-species conservation of a cognitive trait. *Int. J. Comp. Psychol.*, 24:36–59.

Miikkulainen, R., Bednar, J. A., Choe, Y., and Sirosh, J. (2005). *Computational Maps in the Visual Cortex*. Springer, New York.

Rajagopalan, P., Rawal, A., Miikkulainen, R., Wiseman, M. A., and Holekamp, K. E. (2011). The role of reward structure, coordination mechanism and net return in the evolution of cooperation. In *Proceedings of the IEEE Conference on Computational Intelligence and Games (CIG 2011)*, pages 258–265.

Rawal, A., Rajagopalan, P., and Miikkulainen, R. (2010). Constructing competitive and cooperative agent behavior using coevolution. In *Proceedings of the IEEE Conference on Computational Intelligence and Games (CIG 2010)*, pages 107–114.

Rawal, A., Rajagopalan, P., Miikkulainen, R., and Holekamp, K. (2012). Evolution of a communication code in cooperative tasks. In *Artificial Life*, volume 13, pages 243–250.

Reader, S. M. and Laland, K. N. (2002). Social intelligence, innovation, and enhanced brain size in primates. *PNAS*, 99(7):4436–4441.

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

Stanley, K. O. and Miikkulainen, R. (2004). Competitive coevolution through evolutionary complexification. *J. Artif. Intell. Res.*, 21:63–100.

Sternberg, R. J. and Grigorenko, E. L. (2002). *The general factor of intelligence: How general is it?* Psychology Press.

Tomasello, M. and Call, J. (1997). *Primate cognition*. Oxford University Press, Oxford, UK.

van der Maas, H. L. J., Dolan, C. V., Grasman, R. P. P. P., Wicherts, J. M., Huizenga, H. M., and Raijmakers, M. E. J. (2006). A dynamical model of general intelligence: The positive manifold of intelligence by mutualism. *Psychological Review*, 113(4):842–861.

Watson, R. A., Palmius, N., Mills, R., Powers, S. T., and Penn, A. (2011). Can selfish symbioses effect higher-level selection? In *Advances in Artificial Life. Darwin Meets von Neumann*, pages 27–36. Springer.

Whiteson, S., Kohl, N., Miikkulainen, R., and Stone, P. (2005). Evolving soccer keepaway players through task decomposition. *Machine Learning*, 59(1-2):5–30.