Proceedings of the Sixth International Conference on Control, Automation, Robotics and Vision (ICARCV'2000), 5-8 Dec 2000, Singapore.

An Incremental Approach in Evolving Robot Behavior

Dhiraj Bajaj, Marcelo H. Ang Jr.*

Department of Mechanical Engineering
The National University of Singapore
10 Kent Ridge Crescent, Singapore 119260
*mpeangh@nus.edu.sg

Abstract

This paper describes an incremental evolutionary approach used in the development of a suitable neural controller for achieving robust obstacle avoidance behavior, which is then further fine-tuned towards a wall following one for a simple mobile robot. The incremental approach mainly involves an alteration of the environment in which the evolution takes place as well the fitness function used in the genetic algorithm. This approach has been seen to be more fruitful than a single direct approach. Interesting behaviors have evolved from this incremental approach.

1. Introduction

One way of defining the intelligence of autonomous mobile robots is in their ability to survive and perform the needed tasks in new and different environments. In recent years, the evolutionary robotics [1] approach has gained much foothold in the area of incorporating this artificial intelligence into robots. Neural controllers have seen to be widely optimized by evolutionary techniques (which work on the Darwinian theory of survival of the fitness). For such purposes, genetic algorithms (GA) have been used [3,6,7] and have been proven to be successful. However, such techniques require careful planning from the control algorithm's point of view and relating that with the robot as an agent to be evolved.

2. Incremental approach

In order to achieve certain robot behaviors using an uninterrupted uniform GA run, a high complexity level might be encountered such as in designing the appropriate fitness function and selecting the environment in which the robot is to be evolved. This level of complexity can be reduced by breaking down the evolutionary process into a few segments and incrementally try to tailor the robot's behavior until a desired performance level for a certain behavior is achieved. Thus, in incremental evolution, the present ongoing GA is now altering a fairly converged solution achieved through the previous stage. In [4], a two-stage incremental approach (first in medium light followed by strong light conditions) was used to simulate the evolution of a robust obstacle-avoidance behavior in a Khepera robot. Inman Harvey points out that in SAGA (Species Adaptation Genetic Algorithms)[2], the initial population is always fairly converged, i.e. the genotype of different members of the population are rather similar to one another, as opposed to conventional GA in which the initial population is made up of random and diverse genotypes widely spanning the multi-dimensional genotype subspace. In performing incremental evolution on mobile robots, a variety of possible strategies can be put in place such as altering the GA parameters, fitness function, physical environment, morphology of the robot, the length of the genotype, architecture of the neural network and genotype to phenotype mappings during the course of the evolution.

3. Our experiment

The mobile robot on which our work was performed is the Khepera miniature robot [8], shown in Figure 1. This robot is widely used in the development and testing of evolutionary techniques due to its many desirable features. It is circular in shape with a diameter of 50mm, height of 30mm and weighing only 70g. In its basic configuration, it has 2 independent DC motors with encoders, 8 infra-red sensors (emitter/receiver), an onboard 68331 microcontroller and an onboard battery. To test our evolutionary algorithms, the experiments were done in simulation using Webots2.0 [9] software package. This software allows us to test our evolutionary control algorithms on a simulated Khepera robot. After testing, the same program can downloaded to the Khepera robot. The use of the simulation alleviates the long time typical to evolutionary algorithms.

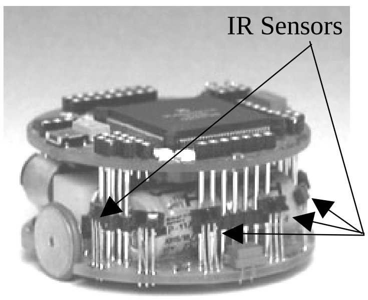

Figure 1: The Khepera robot.

Our aim was to test the incremental approach by first creating a robust neural controller for the mobile robot for straight navigation while avoiding obstacles and later extend it to a wall following behavior. Although obstacle avoidance and wall following behaviors separately, have been well covered by many researchers [3,4,6], we here use the incremental approach to aid in the evolution and thus make the process extremely flexible with this step by step approach. Moreover, we attempt to address some issues in this approach that has come to our attention.

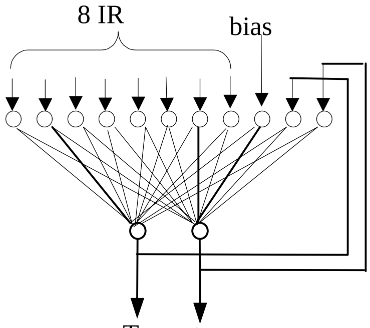

Figure 2: Network used.

A single layer neural network architecture [3] was used with 11 input nodes and 2 output nodes, as shown in Figure 2. 8 of the input nodes were fed by values from the 8 IR sensors of the Khepera working in the active mode (values range from 0 to 1024, where higher values depict closer proximity to obstacles). 2 nodes correspond to the recurrent connections from the output nodes while the last is the bias node with an input value of unity. The sigmoid transfer function was used at the 2 output nodes and scaling was done to achieve ±20 integer values representing a continuum of negative and positive rotational velocity commands to each wheel of the robot. The weights of the network are to be optimized by the GA and a one to one mapping of the genotype to phenotype was implemented. (Please note that the network is not a feedforward type and is not trained using the typical back-propagation algorithm.) The GA used the roulette-wheel-parent selection method and for reproduction it made use of the single point crossover, at probability rate 0.15, and mutation, at probability rate 0.1 for each gene, as the genetic operators. The initial population consisted of 100 individuals randomly generated. Each individual was allowed to run for 192 seconds and assigned a final score thereafter. However, the final fitness score of each individual was an addition of the fitness values after each time-step of 64 ms during its life.

In the first stage, the obstacle avoidance behavior, the fitness formula implemented (adopted from [3], and used in standard ER experiments) for each individual was:

$$f_1 = \sum^{3000} abs(v) * (1 - \sqrt{dv}) * (1 - s)$$

where v is the average rotation speed of both wheels, dv being the difference between the velocity of the wheels and s being the highest IR sensor value. Thus, the 3 components of the fitness function used, rewards high average wheel speed, straight navigation and obstacle avoidance, respectively. The environment in which the robot was placed was made to be a rather simple one, as shown in Figure 3.

In the second stage, the robot was placed in a more complex environment, as shown in Figure 4. The GA being used here starts off with a population of individuals from one of the later generations of stage I, i.e. when the robot was able to navigate. Elitism was also incorporated in the GA in this stage, in which the top 5 percent were retained, while other details of the GA were left unchanged.

In the third stage and last stage, in which the robot's behavior is to be altered from navigating randomly to a wall following one, the fitness function implemented was

$$f_2 = f_1 * (wff)^2,$$

where wff is the wall following factor. It is defined as the fraction of the total time an individual is continually close to a wall on any one side of the robot. Thus the selection procedure rewards individuals that navigate but yet keep fairly close to obstacles while avoiding them.

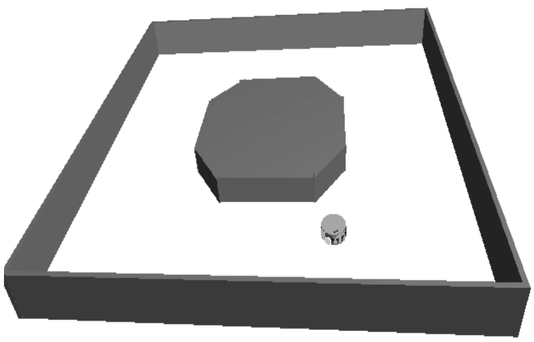
Figure 3: A simple Environment in which the robot was evolved in stage I.

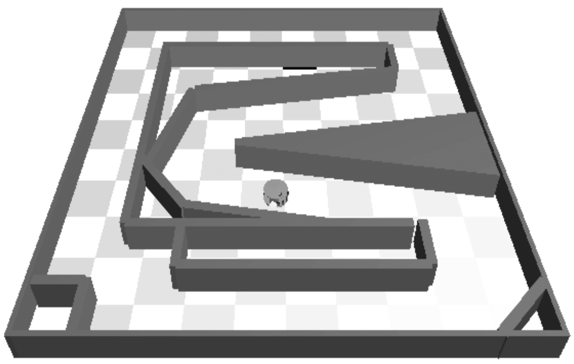
Figure 4: A more complex environment used to fine-tune the robot's behavior.

4. Performance Evaluation

Later generation individuals in stage I performed well, and were able to navigate in their environment while successfully avoiding the walls. The plot of average fitness scores of individuals of each generation against generations is shown in Figure 5, in which the evolution is run for 100 generations. However, when tested in new and complex environments, not all of the evolved individuals that were doing well, performed well. This is thought to be due to the increased complexity of the environment, which the individuals are not used to or not developed sufficiently to perform in.

Thus, in the second stage, the individuals were further evolved in a more complex environment, which had much closer walls and sharp turns. Figure 6 shows the evolution of the fitness function. Here, the new environment had the effect of pressuring the individuals to perform to their maximum ability and thus distinguishing between the really robust individuals and the ones that could just perform simple navigation. In another attempt to force the evolution in the desired direction using the new environment, the best few individuals of any generations would always be brought forward into the next generation. This is known as elitism selection and the balance of the population of the next generation were the offspring of parents selected by the roulette wheel selection procedure from the whole of the earlier generation. This means that the best few individuals are brought over untouched and also further have a chance to be selected again and be in the remaining population. Figure 7 shows the evolution with elitism selection. The result was that individuals of types that could not negotiate the sharp corners and ones that moved in slightly curved paths were eliminated rather quickly. Surviving individuals that filled later generations were ones that strolled comfortably and continuously in the center of the tight path between the two walls. In this stage, after around the first 20 generations, the population was filled up with well performing individuals that displayed excellent straight navigation while avoiding obstacles. Here, it is thought that the GA, aided by the use of a demanding environment, had the primary effect of extracting and filtering the best individuals from the initial pool while the secondary effect being in further fine-tuning the individual genotypes.

In stage III, when the GA used an initial population from the end of stage II, no wall following behavior was achieved, as seen from curve B in Figure 8. This could be attributed to the highly converged population of genotype achieved at the end of stage II and thus the failure of the genetic reproductive operators in producing diverse individuals that were awarded a high fitness score. Also, if the GA is performed with a randomized initial population, no wall following behavior emerges. However, if the GA used an initial population from the end of stage I, a wall following behavior emerged in later generations, and the fitness values are illustrated by curve A in Figure 8. Those individuals moved and kept fairly close to a wall, following it's contour. Thus, by altering the fitness function by adding a wall following term, a new behavior is achieved which still possesses some of the old traits of navigating and avoiding obstacles. It is also interesting to note that once this wall following behavior develops, those individuals lose their ability to navigate in a straight line in the absence of walls or any obstacles. This raises a very important issue that by incremental evolution, how can we not kill off desirable traits of earlier stages that were so difficulty achieved. This remains a research issue and is related to the stability-plasticity dilemma in neural networks.

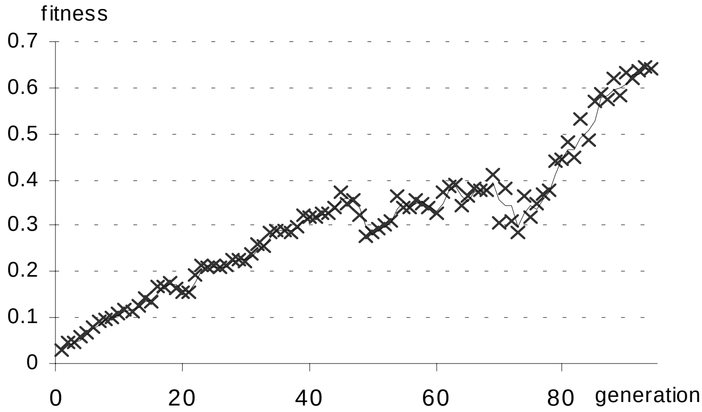

Figure 5: Increasing average fitness values in stage I.

Figure 6: Evolution of average and best fitness value in the complex environment.

In evolving the behavior of an autonomous robot, the physical environment is a very crucial parameter for the successful evolution of its behavior. To understand the importance of the physical environment, the robot should be viewed instead, as an agent interacting with its surroundings, just as what happens in nature. Charles Darwin suggested that species adapt to the environment in the race for survival, which in turn changes their inherent properties, thus giving way to the evolution of new sub-species. Relating this to autonomous robots, the agent must therefore be given substantial opportunity to first learn to survive and be able, then forced to perform and to prove itself to be creative by the use of altering its surroundings.

In the experiments described previously, the surroundings in which Khepera was evolved was static. It is now of interest to test whether the evolution of the required behavior is aided by the use of a dynamic environment instead. A dynamic environment is improvised by including another robot in the maze. Both robots are now being co-evolved, but each without the knowledge of the presence of the other. The environment is made about fifty percent larger in size than the one shown in Figure 3, that to avoid excessive collision between the two robots, which is at least not good in the earlier generations.

The result of the evolutionary run is that, both agents develop obstacle avoidance behaviors. However, no enhancement in performance is noted in the time taken to obtain the desired behavior as compared to the evolution of a single agent alone. Furthermore, very interestingly, it is noticed that in all of the 3 evolutionary runs performed, one agent will always end up substantially fitter than the other agent at its expense. This can be seen by the progressive fitness in Figure 9.

5. Practical Issues

Incremental evolution is basically about altering the problem at stages while using a converged solution. This brings us to the issue of how drastic can the changes at each stage be. Basically, the evaluation procedure of the present evolutionary algorithm must be able to recognize that the converged solution being used is fairly healthy and thus only certain aspects of the genotype will be molded. However, if the evolutionary algorithm does not reward the converged solution being used, and thus instead starts finding new solutions elsewhere in the genotype sub-space by random means, then the purpose of the increment approach has been defeated. This could be possible due to poor selection of a new environment or fitness function, just to name a few, that could make in extremely difficult for the selecting procedure of the algorithm to select the good individuals as they all look equally poor in performance.

Increasing the level of complexity of the physical environment is a classic example in molding robot behavior and has proven very successful in upholding the image of the ‘incremental approach’. As long as few of the individuals in the population can survive in the new complex environment, and the new

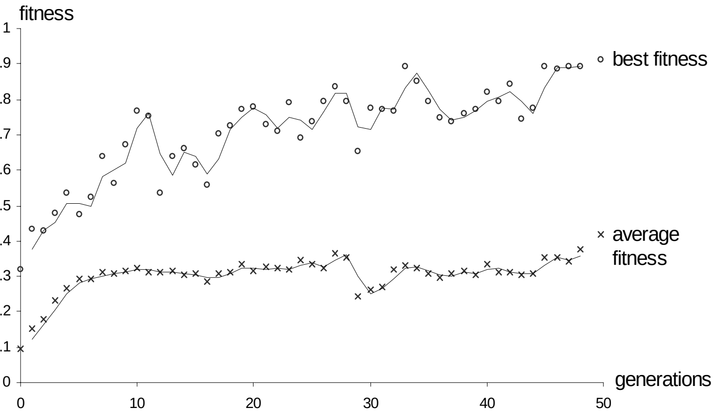

Figure 7: Evolution with elitism in complex environment.

Figure 8: The curve A illustrates the successful development of the wall following behavior when the initial population used is from end of stage I, while curve B illustrates that incremental evolution did not succeed when the initial population was from end of stage II.

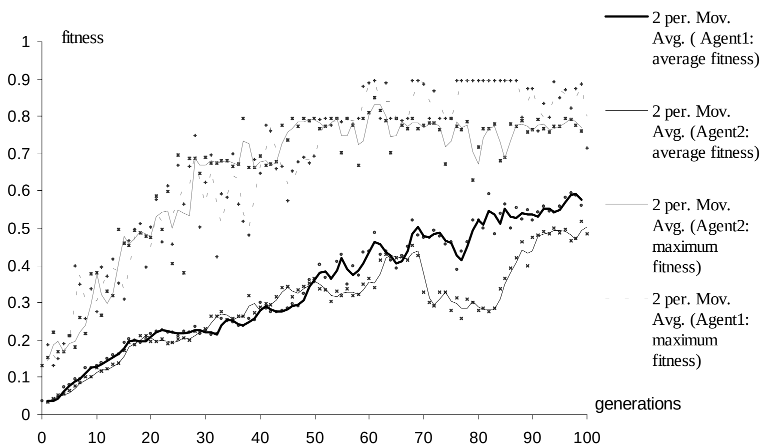

Figure 9: Co-evolution of both agents.

environment provides the opportunity for the needed interactions between the robot and the environment, the new environment should be a good choice. However, the alteration of the fitness function as a tool to alter the problem is much trickier than choosing a suitable new environment. As in the experiment conducted, in stage III, adding only a new term to the previous fitness function successfully altered the fitness function. This way, the new fitness function, having the old components present, can still help in keeping the traits of the individuals developed in the earlier stage. As an illustration, when the wall following behavior developed in stage III, the robot still retained its qualities to avoid banging into obstacles.

Another issue that has come to our attention is the level of convergence of the solutions. As in the experiments carried out, when the evolution is carried over from stage II to stage III, no wall following behavior develops, as it is very difficult to work with a highly converged solution. In [5], Inman Harvey points out that in incremental approaches, recombination (crossover) has a lesser role while the mutation should be made the primary genetic operator. However, for a highly converged solution to be fairly diversified, the mutation rate would have to be increased by a substantial amount, and this would in turn produce much undesired diversity in the population.

6. Conclusion

In the simulation experiments conducted, for the Khepera robot, using a GA to optimize a neural based controller, a simple obstacle avoidance behavior was evolved. This set of results of the simple obstacle avoidance behavior was further used to test the validity of the incremental evolutionary approach. Thus, this led to the achievement of a better, more robust neural controller for straight navigation with obstacle avoidance and a wall following behavior developed for the robot separately. This was achieved due to the alteration of the evolutionary parameters such as the physical environment used for the interaction of the robot and the fitness function while bearing in mind the importance of using a set of solutions that is only fairly converged. While much careful planning and thought is needed in designing the next evolutionary stage in the incremental approach, this approach has shown considerable success in obtaining more complex and detailed behaviors. Finally, with GA being a successful tool in developing autonomous controllers, its extension into an incremental evolutionary approach definitely seems very attractive.

References

[1] D. Cliff, Inman Harvey, and Philip Husbands. *Explorations in evolutionary robotics*. In Adaptive behavior, 1993.

[2] Inman Harvey. *Species Adaptation Genetic Algorithms: A Basis for a Continuing SAGA*. In 'Towards A practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life', 1992, MIT Press.

[3] D. Floreano and F. Mondada. *Automatic Creation of an Autonomous Agent: Genetic Evolution of a Neural-Network Driven Robot*. In From Animals to Animats III: Proceedings of the Third International Conference on Simulation of Adaptive Behavior. MIT Press-Bradford Books, Cambridge, MA, 1994.

[4] Joel Chavas, Christophe Corne, Peter Horvai, Jerome Kodjabachian, and Jean-Arcardy Meyer. *Incremental Evolution of Neural Controllers for Robust Obstacle-Avoidance in Khepera*. In Lecture Notes in Computer Science; Evolutionary Robotics. First European Workshop, EvoRobot98 Proceedings. Springer.

[5] Inman Harvey. *Artificial Evolution for Real Problems*. In Evolutionary Robotics, From Intelligent Robots to Artificial Life, ER97. AAI Books.

[6] Henrik Hautop Lund, Orazio Minglino. *Evolving and Breeding Robots*. In Lecture Notes in Computer Science; Evolutionary Robotics. First European Workshop, EvoRobot98 Proceedings. Springer.

[7] D. Floreno and F. Mondada. *Evolution of Homing Behavior in a Real Mobile Robot*. In IEEE Transaction of Systems, Man, and Cybernetics. PartB. 1996.

[8] http://www.k-team.com/robots/khepera/. The Khepera robot is developed by K-Team S.A., which has accumulated comprehensive experience in the design**,** prototyping and production of the Khepera robot and all its accessories

[9] Cyberbotics, founded in 1998, is the developer of the Webots software, a 3D mobile robot simulator.