Evolutionary Robotics

Jai .
20045247@studentmail.ul.ie

Rohan Arya
20086377@studentmail.ul.ie

Dennis Kolomiyets
20250762@studentmail.ul.ie

Oussoubi Niakate
20242905@studentmail.ul.ie

Abstract:

The main aim for this essay is to educate the reader about the study of evolutionary robotics and certain areas surrounding the topic. This essay will explore the current applications of evolutionary robotics. The essay will discuss how evolutionary (genetic) algorithms can be used to evolve the design and behaviour of a robot and how this approach has been successful in creating robots with novel and adaptive capabilities. Additionally, the essay will examine some open issues of evolutionary robotics with topics such as The Bootstrap Problem and the Reality Gap.

Keywords: Evolutionary robotics, Embodied intelligence, Robotics, Evolutionary Algorithms, Kephera Robot, Bootstrap Problem

Table of Contents

1. Introduction

Evolutionary robotics is a field of study that aims to use the principles of evolutionary biology to design and optimise robotics. “Evolutionary robotics applies the selection, variation, and heredity principles of natural evolution to the design of robots with embodied intelligence” (Doncieux, 2015) In this modern day and age of technology, we have witnessed an exponential rise in the field of automation technology, machine learning, artificial intelligence and robotics. This can be seen in the various kinds of advanced robotic machines that are employed in industries to perform autonomous yet non-adaptive tasks, that execute the same sequence of actions repeatedly without any further human intervention. Similarly, autonomous drones, where the majority of their applications lie in professional photography and cinematography, or for military purposes in field combat, are examples of adaptive yet non-autonomous machines. Although there are new and upcoming machines that have advanced AI systems integrated within, that enable them to emulate autonomous functionality while being able to self-adapt, they still are nowhere near the one force that is capable of being fully autonomous as well as adaptive which is biological evolution.

But when we talk about Evolutionary Robotics, there is more to it than meets the eye. The basic principle of Evolutionary Robotics is based on the selection, variation, and heredity principles of natural evolution that can be applied to the robot’s design accompanied by embodied intelligence, with the main objective to create and design more adaptive, and robust robots. One of the key determining factors that separate Evolutionary Robotics from Robotics is the use of evolutionary algorithms or the concept of one class of population-based-metaheuristics, to optimise the aspects of autonomous machines and robots. It is this use of metaheuristics in this specific field of robotics that sets this field apart from the orthodox research and study in the field of robotics, by integrating advanced machine learning algorithms that are used to control the policy of a robot.

Evolutionary Robotics also opens doors to innovative approaches that can allow us to study the concepts of evolution through newer and advanced experimental tools. All of this can help us to develop a deeper understanding of robotics and to help explore and study concepts that are yet to be investigated through computer simulations and biological sciences.

Artificial Intelligence in itself is a system that emulates the human mind by learning, solving problems and making spontaneous decisions without needing a specific program to be integrated within the system. Since the birth of computers, the goal has been to invent technology that would be capable of reproducing the aspects of the human brain, some examples being natural language processing and deductive reasoning. In the case of Robotics, the basic idea has always been to generate non-cognitive yet adaptive behaviour in robotics that deals with object manipulation. “The idea of using evolutionary principles in problem-solving dates back to the dawn of computers (Fogel, 1998), and the resulting field, evolutionary computing (EC), has proven successful in solving hard problems in optimization, modelling, and design (Eiben and Smith, 2003)." It is this idea of evolution that allows us to design the robot's apparatus, morphology and control simultaneously by implementing a holistic approach to the robot's design that is based on variation and selection principles. If we take a further dive through history, we can see that origin of Evolutionary Robotics dates back to the early 1990s. 'Intelligence without representation' was one of the first books ever published in this field, by Rodney Brooks, that talked about how a newer approach needed to be implemented to shift the focus from rule-based programming in robotics to behaviour-based programming. It was around that time when scientists like Chris Langdon and John Holland were studying evolutionary algorithms to simulate evolution within living systems. The Robots that were being worked once started to simulate the behaviour, the goal was to scale the behaviour-generating algorithm to a level where it starts becoming more complex until we can declare it as intelligent behaviour. "This operational definition of intelligence bears a resemblance to the Turing Test: if a robot looks as if it is acting intelligently, then it is intelligent."(Josh.C.Bongard, 2013)

2. Embodied Evolution and Evolutionary Robotics

While studying and researching about evolutionary robotics, we also use the concepts of Embodied Evolution and how it can be implemented. This can be done by developing a large population of robots that are employed in applications of evolutionary robotics, and using evolutionary algorithms. "Embodied evolution requires a larger number of robots than that used in any evolutionary robotics work to-date. The short-term proof-of-concept experiments (described in the next section) require only minimal capabilities of each robot. Similarly, the long-term objectives of EE emphasize the interaction of robots rather than the sophistication of individual robots."

In embodied evolution, the morphology of the robot can be studied as genotype, where its evolution process takes place using evolutionary algorithms. This evolution can be the robots' size, shape, amount of machinery and other factors such as agility and elasticity. With this we can achieve robotic systems that not only mimic, but as the name suggests, embody real life organisms such as hexapods and quadrupeds. In doing this we get the benefit of studying and creating robots with distinctive morphologies that can perform newer and advanced tasks, which can help us to study the new patterns in Evolutionary Robotics while making discoveries in biological sciences. Embodied Evolution has truly helped us to create and study robots that are better suited to adapt their environment and perform autonomously.

3. Evolutionary Algorithms

"Evolutionary algorithm (EA) is an umbrella term used to describe population-based stochastic direct search algorithms that in some sense mimic natural evolution" (Bartz‑Beielstein, 2014). Evolutionary algorithms are the computational methods that are inspired by biological evolution. These algorithms are based on the principle of evolutionary computation, that is a generic metaheuristic optimization algorithm where these algorithms are integrated within robotics to solve problems over time in various applications such as fitness and problem-solving ability or both. The idea around which evolutionary algorithms revolve around is the creation of the population of robots or control systems that can be of random and predefined variations. This can be used to evaluate their fitness levels based on certain parameters that decide their ability to solve problems and perform tasks, like selection, crossover and mutation.

It is important to note that the most crucial components of evolutionary algorithms are the autonomous process of reproduction and evaluation between the robots in order for Embodied Evolution to be successful. Which is why it is critical for a metric to be coded in the robot as it is autonomous. “This can be quite implicit, for example, where failing to maintain adequate power results in “death” [Mondada and Floreano, 19961. Or”, it can be explicitly hard-coded, for example, where fitness is a function of objects collected and time(R.A. Watson, 1999)”.

4. Experiments in Evolutionary Robotics

Photoaxis Task

There are multiple experiments that have been performed in the field of evolutionary robotics. One such example is the Photoaxis Task. In this task, we study the behaviour of the robot by while a circular light is placed right above it. The behaviour of this robot is simulated using a neural network architecture. The main of the robot is to follow the light source and move along in the direction where the lamp goes. There is an infra-red beacon that is mounted right besides the source of light that monitors the movement of the robots. Once the robot reaches the light source, it triggers a behaviour within the robot that reorients it and once again follows the light source. This happens due to the genetic algorithms that the researchers integrate in the robot in order to evolve the neural network that allow the robot to be mobile by identifying the light source, thus making it adaptive and automated.

Similarly, there is the control architecture, that deals with the robot’s control system and its architecture. The genetic algorithms have been implemented in these robots to evolve their neural network controllers so that they can perform other tasks like avoiding obstacles, following goals and navigating walls. The advantage of performing this algorithm is that we are able to establish that the result of evolved architecture was much better than pre-designed architecture.

Maintaining Reproductive Energy Level

“Energy levels regulate reproduction events and should reflect the robot’s performance at the task.”(R.A Watson, 1999). Another challenge that we assess in evolutionary robotics is the one where we try to maintain the reproductive energy levels of the robots. This can be done by using one of the previous tasks, such as seeking a light source or avoiding obstacles. Once the robot is performing the particular task, we study its reproductive energy levels. The genetic algorithm that is integrated in the robot optimizes the robot’s behaviour in such a way, which not only allows it to evolve itself to solve problems and tasks, but also enables it to sustain its energy levels while navigating and adapting to its environment.

By performing these experiments, we are able to study and observe the potential and ambidexterity evolutionary robotics and are able to compare the results and advantages after integrating genetic evolutionary algorithms within these robots. With this the robots are able to self-learn and self-adapt to the real-world environment that will help us to develop robots that can perform these real-work tasks and solve these problems without any human intervention.

```
procedure evolve
    initialize_genes                  //all weights start at 0
    Energy = MinEnergy                //min_energy is 10
    repeat forever
        do_task_behavior
        Energy = min(MaxEnergy, Energy + get_reward)
        if (receive_gene? and random(MaxEnergy) > Energy) then copy_received_gene
        if (random(MaxEnergy) < Energy) then send_a_gene
    end repeat
end procedure

procedure do_task_behavior
    do_task_specific_behavior         //read sensors, calculate outputs, update motors, check if stuck
    do_task_specific_evaluation       //check for success beacon (and return to random position in
                                      pen)
end procedure

procedure copy_received_gene
    write_received_value_over_corresponding_extant_value
end procedure

procedure send_a_gene
    pick_gene_at_random               //one of four weights with equal probability
    mutate_value                      //add {-1,0,1} with equal probability
    broadcast_mutated_value           //on local infra-red (don't store the mutated value)
    Energy = max(MinEnergy, Energy - constant)
end procedure
```

5. Applications and Impact of Evolutionary Robotics

5.1 Khepera Robot

The Khepera robot is a popular autonomous robot that is commonly used in robotics research. It was developed in the early 1990s by a team of researchers led by Prof. Francesco Mondada at the Ecole Polytechnique Federale de Lausanne (EPFL) in Switzerland. Francesco Mondada is a swiss professor in artificial intelligence and robotics and has a masters degree in Microengineering. He also directed the design of the S-bot, the e-puck, the marXbot and the Thymio mobile robots.

The first robot developed was a small multi-wheeled robot about 5 cm in diameter. First developed in 1996. The Khepera robot was developed for research purposes in the field of robotics. Specifically, it was designed as a platform for exploring various aspects of swarm robotics, which involves the study of how groups of robots can work together to achieve a common goal. Swarm robotics is a different field from evolutionary robotics, where the behaviour of the swarm is determined by the interactions between the robots themselves and their environment, without an external controller. This doesn’t mean that the two fields aren’t related in any way. There is some overlap for example, evolutionary techniques can be used to optimize the behaviour of a swarm and on the other hand swarm robotics can be used to test evolutionary robotics algorithms. An example of an evolutionary robotics technique that was used to develop the Khepera robot is called Neuroevolution or atleast a variant of it. The researchers in Stefano Nolfi and Dario Floreano’s paper Evolutionary robotics a review mention how a Neural Network (NN) learning algorithm, which is a training method used to teach a robot to perform a particular task. They exist as a method of trying to mimic the function and structure of the human brain and are created using a series of interconnected nodes that process and transfer information. Similar to the human brain, when a pattern is favourable it strengthens the connection between the nodes, based on feedback where the outcome of the experiment is preferable. By strengthening the preferable nodes and weakening others, in time this will allow the neural network to perform its function more accurately and efficiently.

This can be typically achieved using a type of optimization algorithm. Popular optimization algorithms include Adam(this one uses adaptive learning rates and momentum to better the weights contributing to a neural network), AdaDelta (an optimization algorithm that adapts the learning rate based on a moving average of the past gradients and updates.) and Adagrad (which optimises based on the adaptation of learning rates for each weight based on previous history of gradients). This is just a few that can be found in Sebastian Ruder’s article on gradient descent. While researching this I had a question of just what is a gradient in the context of evolutionary robotics.

Gradient is the slope of a function with respect to its parameters, those being its weights and biases. An interesting analogy I found online was comparing a gradient to a map that informs the person how to descend from a hill in the quickest way. When taking neural networks into mind, the gradient tells us how to adjust the network's parameters, which leads to a more efficient outcome. Like the map it helps us get to the bottom of the hill quicker. Hence the gradient tells us how to get a better result, with less errors quicker.

5.2 Robotic Fish

In 2012 the University of Essex in England introduced a new environmental monitoring robot. This robot was in the shape of a fish and the project was led by Dr. Huosheng Hu, who is a professor there. These fish were created for environmental monitoring specifically in order to detect pollution in waterways. Traditional methods of monitoring water pollution can be time-consuming, expensive, and sometimes even dangerous for humans. So the team at the University of Essex developed a robotic substitute in the form of a robotic fish, that can simultaneously observe pollution rates and not disturb the local habitat.

The robotic fish are equipped with sensors that can detect changes in water quality, such as the presence of pollutants or changes in temperature and acidity levels. They are also capable of navigating through waterways and communicating with each other, allowing them to work together to cover a larger area more quickly and accurately. This is another example of using the swarm robotics limb of the robotics tree.

It was developed using both reinforcement learning and an evolutionary algorithm to optimise. They used a technique called Q-Learning to teach the fish how to behave like a real fish.Now Q-learning isn't an evolutionary robotics technique, it falls under machine learning, however it is important to mention it. The way it is tied to evolutionary robotics is due to its use of an evolutionary algorithm to increase the optimisation for the fish's swimming. Though which techniques were used exactly were not specified. What was actually brought up was the process. The researchers used a genetic algorithm to optimise the fish's morphology and swimming behaviour. They began by defining a set of parameters that would affect the fish's performance, such as the shape of the tail fin and the frequency and amplitude of its swimming movements. They then created a population of fish with randomised parameter values and simulated their swimming behaviour. The fish that performed the best in the simulation were selected as the parents for the next generation of fish, and their parameter values were used to create new offspring fish with slight variations. This process was repeated over many generations, with the best-performing fish being selected as the parents for the next generation, until a final optimised design was produced.

5.3 Evolvable Walking Robot

The Researchers at the Norwegian University of Science and Technology in Trondheim developed a robot. This robot is a quadruped that is designed to learn and evolve its walk over time through using an evolutionary algorithm. This robot was first presented in 2013 at the IEEE Congress on Evolutionary Computation. Each of the four legs has 4 degrees of motion, which gives it a high range of motion and allows the machine to move in a variety of directions. The robot's walking gait is controlled by a neural network, which is trained using an evolutionary algorithm to optimise the robot's movement.

The evolutionary algorithm used in the Evolvable Walking Robot project is a variation of the Covariance Matrix Adaptation Evolution Strategy (CMA-ES), which is a popular method for optimising continuous nonlinear functions. The CMA-ES algorithm is used to evolve the neural network controlling the robot's walking gait, allowing the robot to adapt to different terrains and environments over time. A nonlinear function is one where the output of the function does not have a proportional relationship with its input. Examples of non-linear functions include exponential functions, logarithmic functions, and trigonometric functions such as sine and cosine.

The CMA-ES technique was designed to solve optimization problems where the equations to solve are expensive to evaluate. From what I gather it works by maintaining a distribution of probabilities and then using that distribution to create new potential solutions. This distribution then gets updated each iteration based on previous performance and the newer iterations better solutions are determined when the new6 distribution is more focused. These are the most interesting applications of evolutionary robotics and the techniques used that I could find. However, that doesn't mean there aren't any issues with them.

6. Open Issues of Evolutionary Robotics

Despite Evolutionary Robotics having obtained much success in recent years, it has encountered many major issues that have proven to be very difficult to overcome and vital for further development in the field. These issues include:

6.1 The Bootstrap Problem

The bootstrap problem is often recognized as one of the main challenges of evolutionary robotics; the reason being that if all individuals from the first randomly generated population perform equally poorly, the evolutionary process won't generate any interesting solutions.
In order to overcome this lack of fitness gradient, Jean-Baptiste Mouret et al.(2009) proposed to “efficiently explore behaviours until the evolutionary process finds an individual with a non-minimal fitness.” In order to achieve the goal, they introduced an original diversity-preservation mechanism, called behavioural diversity. According to IEEE, it relies on a

distance between behaviours rather than genotypes or phenotypes and multi-objective evolutionary optimisation. This approach has been successfully tested and compared to multi-subgoal evolution, on the evolution of a neuro-controller for a light-seeking mobile robot. The results obtained from these two approaches were qualitatively similar, although the introduced one is less directed than multi-subgoal evolution.

6.2 The Reality Gap

This is another major issue when it comes to Evolutionary Robotics. The reality gap happens when controllers evolved in a simulation prove to be ineffective on physical robots. One suggested solution to this problem by Miglino et al.(1995) is to use samples from the real robots' sensors during simulation. Another one is the transferability approach by Koos(2013), in which the goal is to learn the differences between simulation and reality which can coerce behavioural evolution. One way to eliminate the reality gap is to rely exclusively on real robots for controller evolution, which is extremely time-consuming at the current state of development.

6.3 Genomic Encoding

Genomic encodings and genotype-phenotype mappings are important for enabling the evolution of complex structures. According to Nelson et al.(2009), the vast majority of studies in Evolutionary Robotics utilise direct encoding. Because of this, genotypes directly specify a phenotype and leads to scalability issues due to the fact that each parameter is encoded and optimised separately. On the other hand, indirect encodings allow solutions to be represented as patterns of parameters rather than representing each parameter individually.However, indirect encodings tend to be biassed towards regular structures which makes it difficult for them to notice any abnormalities, such as faults in the joints of a robot.

6.4 Time it takes to Evolve Controllers Directly on Robots

One fascinating solution that could potentially solve this issue according to Scholarpedia, is embodied evolution. This is the process of distributing the evolutionary algorithm across a group of robots that evolve in parallel and exchange genetic information, seeding the evolutionary process with pre-evolved or pre-programmed partial or approximate solutions. Or the onboard fusion of simulation-based evolution and online evolution, where each robot keeps models of its surroundings and of other robots, and the models are modified in response to variations in controller performance between the onboard simulation and reality. However, the performance benefits of such approaches depend on the encounters between robots, which may be infrequent in large or open environments, on the size of the collective, and on the communication capabilities of the robots.

6.5 Nature Like Evolvability

Perhaps one of the most obvious and challenging issues that Evolutionary Robotics is facing, is the ability to produce robots that naturally evolve over time just like biological organisms. This is difficult because in order to make evolution possible, random mutations need to reasonably happen to produce non-lethal phenotypic variations. Since machines are only excellent at executing tasks they have been given, it's not hard to see why evolution is a difficult feat for them to achieve.

6.6 Combining Evolution and Learning

Unfortunately, mixing evolution and online learning has proven difficult. According to Stephane Doncieux, "many papers about learning in ER address different challenges while using the same terminology (e.g., "learning," "robustness," and "generalisation"), making it difficult to understand the literature."Robots have shown to maintain the same qualitative behaviour(Quality of work/attitude towards a task) despite any environmental/morphological changes. This is due to the fact that there is no explicit reward/punishment system involved to produce "reward-based behavioural changes".

6.7 Evolutionary Robotics and Reinforcement Learning

Evolutionary Robotics and Reinforcement Learning may be inspired by different principles, but they both tackle a similar challenge. A few papers compare the results of using evolution-inspired algorithms and reinforcement learning methods to solve reinforcement learning problems in robotics. Evolutionary algorithms like CMA-ES(Hansen and Ostermeier, 2001) are good optimisers that can be used to optimise the parameters of a policy in lieu of gradient-based optimisers. In a recent series of benchmarks, Heidrich-Meisner(2008) compared many ER and RL algorithms to CMA-ES in evolving the weights of a neural network (Heidrich-Meisner, 2008; Heidrich-Meisner and Igel, 2009). They used classic control problems from RL (cart-pole and mountain car) and concluded that CMA-ES outperformed all the other tested methods. Stulp and Sigaud (2012) compared CMA-ES to policy search algorithms from the RL community and concluded that CMA-ES is a competitive algorithm to optimise policies in RL (Stulp and Sigaud, 2012, 2013). Taylor et al. (2006, 2007) compared NEAT (Stanley and Miikkulainen, 2002) to SARSA in the keep-away task from RoboCup. In this case, the topology of a neural network– that is, the structure of the policy – was evolved. Their results suggest that NEAT can learn better policies than SARSA, but requires more evaluations. In addition, SARSA performed better when the domain was fully observable, and NEAT performed better when the domain had a deterministic fitness function.

7. Conclusion

In conclusion, the field of evolutionary robotics is rapidly evolving. The future is very promising regarding the designing and optimising of new robots. Up until now, researchers have been able to create robots with novel and adaptive capabilities through the help of genetic algorithms and the principles of evolution. Capabilities such as locomotion, navigation and complex problem-solving have been achieved. The current issues and challenges the study faces are being met with equally rigorous innovative research and development. Ultimately, evolutionary robotics offers a powerful for advancing the design and development of robotic systems and the future is very exciting.

Sources:

“Evolutionary Robotics” by Fernando Silva et al.
http://www.scholarpedia.org/article/Evolutionary_Robotics#Open_issues_in_evolutionary_robot_engineering

https://www.frontiersin.org/articles/10.3389/frobt.2015.00004/full

https://www.researchgate.net/publication/226687029_Evolutionary_robotics-A_review

"An overview of gradient descent optimization algorithms" by Sebastian Ruder
https://ruder.io/optimizing-gradient-descent/

Li, J., Li, Y., Li, Z., Wang, J., & Li, X. (2014). Design and control of biomimetic robotic fish. International Journal of Advanced Robotic Systems

Glette, K., Zhang, H., Haasdijk, E., & Eiben, A. E. (2013). Evolvable walking robot controlled by an artificial genome. https://ieeexplore.ieee.org/abstract/document/6557873

https://arxiv.org/abs/1604.00772

Doncieux, S., Bredeche, N., Mouret, J.B. and Eiben, A.E., 2015. Evolutionary robotics: what, why, and where to. *Frontiers in Robotics and AI*, *2*, p.4.

R. A. Watson, S. G. Ficiei and J. B. Pollack, "Embodied evolution: embodying an evolutionary algorithm in a population of robots," Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406), Washington, DC, USA, 1999, pp. 335-342 Vol. 1, doi: 10.1109/CEC.1999.781944.

Bartz‐Beielstein, T., Branke, J., Mehnen, J. and Mersmann, O., 2014. Evolutionary algorithms. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *4*(3), pp.178-195.

Josh C. Bongard. 2013. Evolutionary robotics. Commun. ACM 56, 8 (August 2013), 74–83. https://doi.org/10.1145/2493883