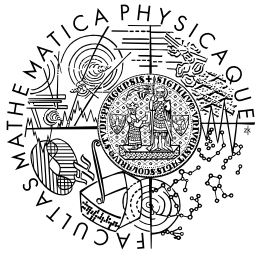

FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

František Dostál

Fitness and novelty in evolutionary reinforcement learning

Name of the department

Supervisor of the master thesis: Mgr. Roman Neruda, CSc.
Study programme: Mgr.
Study branch: Artificial Inteligence

Prague 2023

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In ..... date ....

Author's signature

Dedication.

Title: Fitness and novelty in evolutionary reinforcement learning

Author: František Dostál

Katedra teoretické informatiky a matematické logiky: Name of the department

Supervisor: Mgr. Roman Neruda, CSc., department

Abstract: Novelty is a novel aproach to modeling selection criteria in evolutionary algorithms and has been proven as viable technique of avoiding pitfalls of false optima in tasks abundant with them, such as solving mazes. Rather than closing the topic however, this finding opened other problems to explore: How to properly apply novelty in tasks that yield slightely better to conventional aproaches? How to properly model behavioral space nessessary for novelty computation? In this thesis we investigate use of novelty in selected reinforcement learnig tasks, it's combinations with classical fitness and propose behavior space models for the respective RL tasks.

Keywords: evolution novelty fitness behavioral space

Contents

Introduction

1. Introduction to Artificial Agents

To understand the reasoning guiding this work we have to first look at the fundamental subject of our efforts - the artificial agents.
We will look at their structure, guiding principles, practical implementation and nor last nor least, their use.

1.1 Agents and enviroment

All agents exists in enviroments that provide context to their agency. An agent gets information about the **enviroment** it find itself in (e.i. **percepts**) through **sensors** and acts on the enviroment through **actuators**. Both actuators and sensors are usually characteristics of the agent. Another characteristic of an agent would be setup of its internal machinations.

Definition 1. *[Russell and Norvig, 2021] Let E be enviroment where agent G is located. G is equiped with actuators A and sensors P and so we can identify $obss_P(E)$, a space of all possible sequences of percepts available to G within E and $A(E)$ space of available actions in E. Then function $G : obss_P(E) \rightarrow A(E)$ is called an agent function, defining behavior of the agent G in enviroment E.*

We ilustrate relationship of these concepts in the diagram 1.1. Because we want our agents to solve some tasks in their respective enviroments with certain efficiency i.e. we want them to be intelligent, we need to define a notion of rationality that can help us better capture what we mean by intelligence. First however we need to measure how usefull or detrimental a situation can be for our agent.

Definition 2. *[Russell and Norvig, 2021] Be that E is an enviroment and $seq(E)$ is a space of all possible sequences of states within E. Then function $f : seq(E) \rightarrow \boldsymbol{R}$ is a performance measure of some agent operating in E.*

Definition 3. *[Russell and Norvig, 2021] For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has.*

We should note that in the real-world applications the rationality of an agents can often be limited by their capacity to store the evidence gathered from percepts, by the lack of their built-in knowledge or the computational capacity to

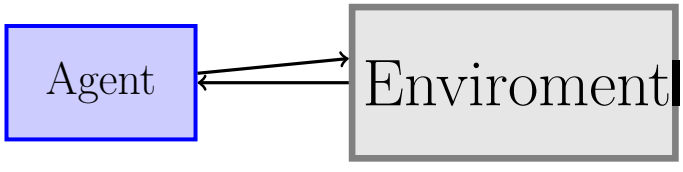

Figure 1.1: Agent and enviroment realtionship

estimate action sequence that can be expected to achieve global maximum of the agent's performance measure.
Since our focus is mainly on the evulationary algorithms in the scope of this thesis we will only encounter the most simple of agents subject to the most severe of such limitations.

Definition 4. *[Russell and Norvig, 2021] An agent is called simple reflex agent if it takes into account only the current percepts, never keeping any history or state information during it's run.*

1.2 Neural Network Agent

The concept of neural networks (NN) comes from an area of machine learning called deep learning that was concieved to attempt to model biological neuron activity. Since it has become one of the most successful branches of machine learning. [Russell and Norvig, 2021] The neural networks consist of neurons organised in layers.

Definition 5. *Let $W \in R^{mxn}$ and $b \in R^n$ and let $f : R^n \to R$.Then we call W the weights of a neuron layer with n neurons, b the bias of the layer and f the activation function of the layer when given an input vector $x \in R^m$ we calculate the output of the layer as $f(W^T x + b)$. Two layers have directional connection when the output of first layer is used as input of the subsequent layer.*

We recognise several typical activation functions:

- sigmoid
$$\sigma(x) = \frac{1}{1 + e^{-x}}$$
- tanh
$$\tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$$
- ReLU
$$ReLU(x) = max(0, x)$$

Definition 6. *[Russell and Norvig, 2021] A neural network where the connections between layers of neural network form an acyclic graph is called a feed-forward neural network.*

Because the feed-forward NN doesn't hold any state or holds onto any previously acquired results, only produces output for a given input, they are ideal for implementation of simple reflex agents.

2. Enviroments

2.1 Markovov Decision Problem

2.2 Reinforcement learning problem

2.3 Open AI Gymnasium

2.3.1 Cartpole

Cartpole is one of the most known benchmark enviroments in the Gymnasium library. It's a simple enviroment where the agent controls a cart rolling on flat surface with a pole on top. The goal is to keep the pole balanced on top of the cart. The reward is calculated every step dpending on the angle

2.3.2 Lunar Lander

Lunar Lander is somewhat more complex domain to optimise for than the Cartpole. Every step the reward is calculated based on several criteria, as oposed to only an angle of a stick, controling the landers descend and additional points are awarded for completing certain tasks. Additionally there are several variables determining the enviroments behaviour during evaluations such as the strenght of wind Unlike in cartpole where

3. Evolutionary algorithms

While reinforcement learning is a useful method of developing expert agents, it has certain drawbacks. Since it only iterates on handful of agents (usually a single agent) with the feedback directly recieved, it does not search the agent space very thoroughly and the resulting adapted agent can be ill-prepared for conditions arising from its good performance in the previous stages of more structured enviroments. If this might be a concern, more suitable way of producing expert agents, might be an evolutionary algorithm. There the agent is typically evaluated on the complex of its behavior within the enviroment.

Definition 7. *An objective function is a mathematical expression that defines the goal of an optimization problem, representing the quantity that needs to be maximized or minimized.*

The evolutionary algorithms are a class of optimalisation methods modeled after our idea of biological process of evolution. Subjects of these optimalisation algorithms are called individuals, often representing feasible solutions of the optimalisation problem. The evolutionary algorithms are specific in that they search the space of individuals semi-randomly using folowing operations:

- recombination (or cross-over)
- mutation
- selection

The selection process usually utilises some kind of value assigned to an individual. Because the purpose of this thesis is to compare performance of different ways to assign such value, some newer than others, we have to sidestep the established terminology in vast majority of the literature a little bit.

Definition 8. *Let A be an evolutionary algorithm with objective function $g : I \to R$ searching over space of individuals I. Then any function $f : I \to R$ used to guide selection in A would be called utility function. Only in the special case where $g = f$ we will call it fitness function.*

3.1 Utility functions

3.1.1 Fitness

Using objective function as the utility function of the algorithm is the most direct way of establishing relationship between the solution we want and the individuals in the evolving population.
Though this aproach ensures that the most optimal individuals from given population will generally be selected into the next generation during the evolution it does not guarantee that their potential for evolving is not diminished either. This way the evolution using fitness can easily get stuck in local optimum of the objective function. While in most cases it might be still possible to reach global optimum on the given problem by changing the hyperparameters to favor random exploration, it may come to point where the benefits of evolutionary aproach are made void.

3.1.2 Pure Novelty

In the paper **?** a new aproach to utility functions has been proposed, called novelty, designed to help avoid traps of local optima.

Definition 9. *[Doncieux et al., 2019] Let E be enviroment and S be a space of all possible states of the enviroment and let $d(x, y)$ be some metric. Then (B, d) is a behavioral space and a function $o_B : S^T \to B$ is called observer function.*

Definition 10. *[?] Let P be a population of individuals within EA and (Img(B), d) be a behavioral space. Then for each individual a we define novelty as such:*

$$N(a) = \frac{\sum_{b \in P, b \neq a} d(a, b)}{|P| - 1}$$

With novelty as utility function the selection process incentivises behaviours inversely to their representation in the current population, in other words according to how "novel" they are. This allows the evolutionary search to retain variety within the population across generations, leveraging the recombination as well as mutations to leave local optima.
Downside of this aproach is in the ambiguity in the choice of behavioral space. Incorporating too many enviroment variables will lead to the curse of dimensionality. Leaving out an important variable means the algorithm will not retain diversity where it matters, making novelty pointless. This means the behavioral function and space have to be selected empirically for given problem.

3.1.3 Combinations of Novelty

The novelty does not take into account the individual's performce with objective function. Therefore when two invidials with similair behavioral characteristics appear but one of them performs significantly better than the other, pure novelty algorithm can easily select the less performant individual, losing the more performant genom to further generations. To prevent this we want to meaningfully combine novlety with fitness.

3.2 Continuous Optimalisation

3.2.1 Evolutionary Strategies [Beyer and Schwefel, 2002]

Evolutionary strategies are

3.2.2 Differential Evolution

4. Comparison criteria and conditions

To compare different types of guiding functions well, we have to select criteria by which we compare them and establish conditions under which will this comparison take place so that our conlusions are fair but not misleading.

4.1 Criteria

Since fitness provides direct measure of performance we would like to make our comparison on the fitness of the final population and its distribution in it. To that end we have to perform experiments with variety of initialisations of the respective enviroments, getting more comprehensive picture then from a single run. This will be mainly reflected in the seed for random numbers generator of the enviroment. Since the promise of the novelty aproach lies mainly in avoiding the trap of local optima we also want to review statistics of each generation produced by the algortihm runs
However we also have to take into account the hyperparameters with which these populations have been achieved. This is especially (but not limited to) population size and number of generations, hyperparameters responsible for most of the computational complexity of a evolutionary algorithm run.

4.2 Conditions

The conditions are defined by the hyperparameters of the algorithms and the hyperparameters of the individuals.

4.2.1 Individual hyperparameter selection

The hyperparameters characterising individuals are number of layers and the number of neurons in their layers.

4.2.2 Algorithm hyperparameter selection

The algorithm hyperparameters will be selected in a series of grid searches, tailored to each enviroment and algorithm type. While it is theoretically possible to setup one large comprehensive gridsearch for each algorithm, lacking specialised hardware, the exponential dependence on number of hyperparameters would make it higly impractical in reality.
Therefore we setup multiple grid searches in logical sequence and groupings so that hyperparameters

5. Experiments

Conclusion

Bibliography

Hans-Georg Beyer and Hans-Paul Schwefel. Evolution strategies - a comprehensive introduction. *Natural Computing*, 1:3–52, 03 2002. doi: 10.1023/A:1015059928466.

Stephane Doncieux, Alban Laflaquière, and Alexandre Coninx. Novelty search: a Theoretical Perspective. In *GECCO '19: Genetic and Evolutionary Computation Conference*, pages 99–106, Prague Czech Republic, France, July 2019. ACM. doi: 10.1145/3321707.3321752. URL `https://hal.science/hal-02561846`.

Stuart Russell and Peter Norvig. *Artificial Intelligence, Global Edition A Modern Approach*. Pearson Deutschland, 2021. ISBN 9781292401133. URL `https://elibrary.pearson.de/book/99.150005/9781292401171`.

List of Figures

List of Tables

List of Abbreviations

A. Attachments

A.1 First Attachment