# Evolutionary Computation

**2023/2024**

## Walking on Thin Ice

**Nuno Lourenço**

**João Correia**

March 5, 2024

# 1 Introduction

We present here the practical project, part of the students' evaluation process of the Evolutionary Computation course of the University of Coimbra. This work is to be done autonomously by a group of **two** students. The deadline for delivering the work is **13 of May** via Inforestudante.

The quality of your work will be judged as a function of the value of the technical work, the written description, and the **public defence**. All sources used to perform the work (including the code) must be clearly identified. The document may be written in Portuguese or in English, using a word processor of your choice[1]. The written report is limited to **10** pages long, but in special, justified, cases (e.g., the need of presenting many images and/or tables), that number may be increased accordingly. The document should be well structured, including a general introduction, a description of the problem, the experimental setup, the analysis of the results, and a conclusion. The report should follow the Springer LNCS format. The Latex and Word templates are available in the Support Material of the course. The final mark will be given to each member of the group individually.

To do the work the student may consult any source he/she wants. Nevertheless, plagiarism will not be allowed and, if detected, it will imply failing the course. While doing the work and when submitting it, you should pay particular attention to the following aspects (whose relative importance depends on the type of work done):

# 2 Problem Statement

In this work we are going to develop an Evolutionary based approach to the Frozen Lake problem available in the Gymnasium framework [2]. The problem involves finding a policy that allowing an agent to move from a starting point to a destination across a lake covered in ice, avoiding any openings by traversing the icy surface. In the most challenging version, the lake is made of slippery ice [3], which may cause the player to slide unpredictably instead of moving in the chosen direction.

The game starts with the agent at the $[0, 0]$ coordinate, and the goal is to reach the farthest point of the grid, such as the $[3, 3]$ in the 4x4 map 1. Holes are scattered across the ice. The agent can move until he reaches the goal or fall in a hole.

[1] We strongly suggest the use of LaTeX.

[2] https://gymnasium.farama.org/environments/toy_text/frozen_lake/

[3] We are going to ignore this aspect of the game

The agent has the following possible actions:

0 : Move Left

1 : Move Down

2 : Move Right

3 : Move Up

At each iteration you get information concerning the player's current position using the following form: $current\_row * nrows + current\_col$ with both the row and column starting at 0. If you consider the example of Fig. 1, the player position is the following: 2 * 4 + 2 = 10. Note the that number of positions depends on the map size.

The game stops when the following conditions are verified:

1. The player falls into a hole;
2. The player reaches the goal;
3. The maximum number of steps is reached;

When the game stops, the agent receive receives a reward of **+1** for reaching the goal, and a reward of 0 otherwise (not reaching the goal or falling into a hole).

# 3 Objective

In general terms, the main objective is to develop an Evolutionary Algorithm that is able to discover a policy that allows the agent to move from the starting point to the goal using the minimal number of steps. To fulfill this task, your task entails the following aspects:

- Develop an Evolutionary Algorithm, which requires you to tackle the following design issues:
  - Representation
  - Initialization Procedure
  - Variation Operators (crossover and mutation)
  - Parent Selection Mechanisms
  - Survivor Selection Mechanisms

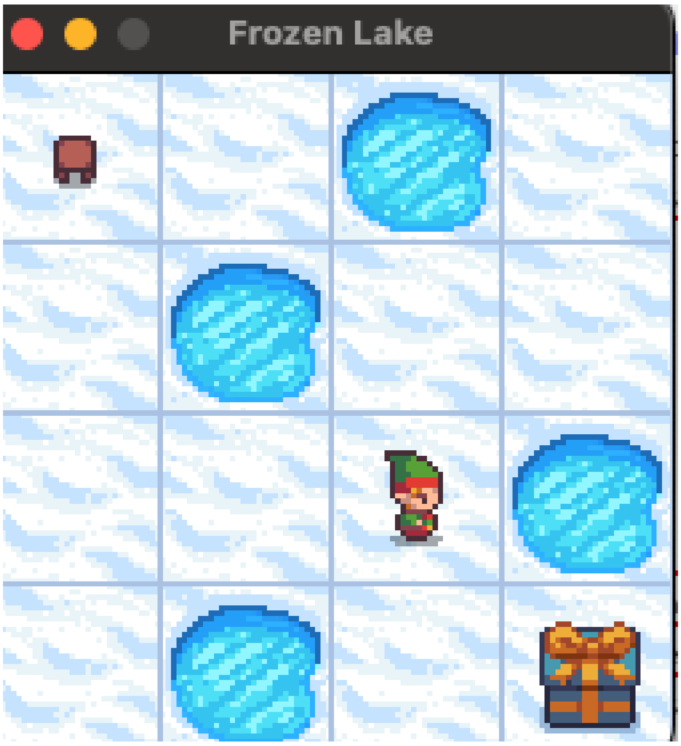


Figure 1: 4x4 Frozen Lake Map Example

  – Fitness Function

As we discussed in the class the fitness function is crucial for the success of the algorithm. You should experiment several fitness functions, taking into account the main goal of the project.

# 4 Experimentation

Regarding the maps to evaluate the proposed solution you should consider the following ones:

- 4x4, with a maximum number of steps (MAX_ITERATIONS_4_by_4) equal to 100;
- 8x8, with a maximum number of steps (MAX_ITERATIONS_8_by_8) equal to 200;
- 12x12, with a maximum number of steps (MAX_ITERATIONS_12_by_12) equal to 500;

We provide some examples of these maps for you to evaluate your solution in the file `maps_to_evaluate.py`. The file also contains the maximum number of steps for each map. However, you can and should test your approach in other configurations. Do not forget to fix the seeds for the random map generator, otherwise you will have a different map every time you execute the algorithm.

When performing the experiments, you should take into account the following points:

- description of the general architecture of the algorithm used;
- description of the representation, variation operators and fitness function used;
- description of the experiment, including a table with the parameters used;
- description of the measures used for the statistical work: quality of the final result, efficacy, efficiency, diversity, or any other most appropriate;
- the minimum number of runs is 30;
- the comparisons must be fair, i.e., you should give to all variants the same resources (number of evaluations);.

Do not forget, besides what was just said, that it is fundamental: (1) to do a correct statistical analysis, including the reasons for choosing a particular method; (2) to do an informed discussion about the results obtained; (3) to put in evidence the advantages of the chosen alternative.

# 5 Conclusion

A few short comments. First, the control of the progression of your work will be done during the classes (T and PL). Moreover, you can discuss eventual problems by presenting yourself during office hours. Second, the projects reflect for the most part your actual knowledge. The rest will be object of lecturing soon after Easter. Third, we try to balance the difficulty of all the work, but we are aware that this is not an easy task and it is somehow a subjective matter. Fourth, we try to ask a workload compatible with the value of the work for the final mark.

Methodological issues, like the statistical background, were elucidated during the previous lectures. You may use the statistical tool you feel at ease with, including the Python code that was provided. Finally, even if this is a work that asks you to do simulations and analyze the results, i.e., it has a practical flavor, there is however a theory behind the work, and you are advised to consult the necessary literature.

Good luck!