General Ideas for the Evaluation Tracker

The given tools to visualize and view the data Open Evolve provides has been deemed not enough, this paper is made to brainstorm key points that should be focused on by our tracker. The tracker will be a program that can receive a control Config, Base Function and Evaluator script and then receive the testing Config, Base Function and Evaluator scripts. These testing files should outnumber the control 10 to 1 minimum and should be unique to the specific variable(s) and/or parameter(s) that are changed.

Variables and Parameters that can be Tracked

Each parameter and variable in the config files, control and testing, should be tracked and compared to one another. This Tracker should output to a general file type like CSV for quick access to compare data. Here are the variables and parameters in the config file that will be changed and their results observed by our supposed tracker.

Config File

General

Max_iterations
Checkpoint Interval
Diff_based_evolution
Max_code_length

LLM

Temperature
Max_tokens
Timeout
Retries

Prompt

System_message
Num_top_programs
Num_diverse_programs
Include_artifacts

Database

Population_size

Archive_size
Num_islands
Elite_selection_ratio
Explotation_ratio
Exploration_ratio

Evaluator Config

Timeout
Max_retries
Cascade_evaluation
Cascade_thresholds
Parallel_evaluations

These are the general ones that seem to span across all examples, the task specific parameters will not be listed here but may be listed on our tracker when it is programmed. Here is the determined purpose of the Tracker relative to the evaluation file.

Evaluation File

The parameters and variables for the effectiveness of changes relative to the evaluation script are problem specific so it probably will not change much between control and testing evaluation scripts. However the tester itself will probably be integrated somewhere in the evaluator script or at least be called following an evaluation resolution.

Here is the determined purpose of the tracker relative to the Base Function

Base Function File

The base function file for testing should vary from the control file in terms of completeness/closeness to known optimum. This should be able to determine the variety of effectiveness compared to the control. This means that for our control and testing group a problem with a known optimum solution would be best for comparison in this category. Although it is important to note that the LLM could have already known the optimal solution to the problem prior to evolving.