Evolutionary Algorithms: Final report

Your Name (r0123456)

November 26, 2025

1 Metadata

- **Group members during group phase:** Group member 1 and group member 2
- **Time spent on group phase:** 10 hours
- **Time spent on final code:** 40 hours
- **Time spent on final report:** 10 hours

2 Changes since the group phase (target: 0.25 pages)

List the main changes that you implemented since the group phase. You do not need to explain the employed techniques in detail; for this, you should refer to the appropriate subsection of section 3 of the report.

1. State here the modification that you made (e.g., replaced top-λ selection with k-tournament selection).
2. …

3 Final design of the evolutionary algorithm (target: 3.5 pages)

Goal: Based on this section, we will evaluate insofar as you are able to design and implement an advanced, effective evolutionary algorithm for solving a model problem.
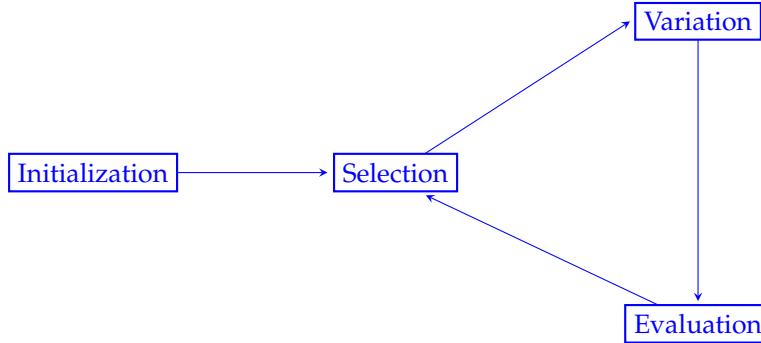
3.1 The three main features

List the three main components of your evolutionary algorithm for this project. That is, what are its most distinctive characteristics, what components am I not allowed to change to a more basic version? Ideally these are some of the more advanced features that you added since the group phase.

1. Just state the feature, you do not need to explain it, instead refer to the appropriate section below.
2. …
3. …

3.2 The main loop

Make a picture of the "flow" in your evolutionary algorithm, similar to the example below. Include all the main components (mutation, recombination, selection, elimination, initialization, local search operators, diversity promotion mechanisms). There are no formal requirements on how to do this, as long as it is clear and you can efficiently explain your complete evolutionary algorithm using this picture at the exam. Contrary to the picture below, include the specific techniques, e.g., top-λ elimination, k-tournament selection, where possible.

The questions from ?? to ?? in blue are there to guide which topics to discuss, rather than an exact list of questions that must be answered. Feel free to add more items to discuss.

3.3 Representation

How do you represent the candidate solutions? What is your motivation to choose this one? What other options did you consider? How did you implement this specifically in Python (e.g., a list, set, numpy array, etc)?

3.4 Initialization

How do you initialize the population? How did you determine the number of individuals? Did you implement advanced initialization mechanisms (local search operators, heuristic solutions)? If so, describe them. Do you believe your approach maintains sufficient diversity? How do you ensure that your population enrichment scheme does not immediately take over the population? Did you implement other initialization schemes that did not make it to the final version? Why did you discard them? How did you determine the population size?

3.5 Selection operators

Which selection operators did you implement? If they are not from the slides, describe them. Can you motivate why you chose this one? Are there parameters that need to be chosen? Did you use an advanced scheme to vary these parameters throughout the iterations? Did you try other selection operators not included in the final version? Why did you discard them?

3.6 Mutation operators

Which mutation operators did you implement? If they are not from the slides, describe them. How do you choose among several mutation operators? Do you believe it will introduce sufficient randomness? Can that be controlled with parameters? Do you use self-adaptivity? Do you use any other advanced parameter control mechanisms (e.g., variable across iterations)? Did you try other mutation operators not included in the final version? Why did you discard them?

3.7 Recombination operators

Which recombination operators did you implement? If they are not from the slides, describe them. How do you choose among several recombination operators? Why did you choose these ones specifically? Explain how you believe that these operators can produce offspring that combine the best features from their parents. How does your operator behave if there is little overlap between the parents? Can your recombination be controlled with parameters; what behavior do they change? Do you use self-adaptivity? Do you use any other advanced parameter control mechanisms (e.g., variable across iterations)? Did you try other recombination operators not included in the final version? Why did you discard them? Did you consider recombination with arity strictly greater than 2?

3.8 Elimination operators

Which elimination operators did you implement? If they are not from the slides, describe them. Why did you select this one? Are there parameters that need to be chosen? Did you use an advanced scheme to vary these parameters throughout the iterations? Did you try other elimination operators not included in the final version? Why did you discard them?

3.9 Local search operators

What local search operators did you implement? Describe them. Did they cause a significant improvement in the performance of your algorithm? Why (not)? Did you consider other local search operators that did not make the cut? Why did you discard them? Are there parameters that need to be determined in your operator? Do you use an advanced scheme to determine them (e.g., adaptive or self-adaptive)?

3.10 Diversity promotion mechanisms

Did you implement a diversity promotion scheme? If yes, which one? If no, why not? Describe the mechanism you implemented. In what sense does the mechanism improve the performance of your evolutionary algorithm? Are there parameters that need to be determined? Did you use an advanced scheme to determine them?

3.11 Stopping criterion

Which stopping criterion did you implement? Did you combine several criteria?

3.12 Parameter selection

For all of the parameters that are not automatically determined by adaptivity or self-adaptivity (as you have described above), describe how you determined them. Did you perform a hyperparameter search? How did you do this? How did you determine these parameters would be valid both for small and large problem instances?

3.13 Other considerations

Did you consider other items not listed above, such as elitism, multiobjective optimization strategies (e.g., island model, pareto front approximation), a parallel implementation, or other interesting computational optimizations (e.g. using advanced algorithms or data structures)? You can describe them here or add additional subsections as needed.

4 Numerical experiments   (target: 1.5 pages)

Goal: Based on this section and our execution of your code, we will evaluate the performance (time, quality of solutions) of your implementation and your ability to interpret and explain the results on benchmark problems.

4.1 Metadata

What parameters are there to choose in your evolutionary algorithm? Which fixed parameter values did you use for all experiments below? If some parameters are determined based on information from the problem instance (e.g., number of cities), also report their specific values for the problems below.

Report the main characteristics of the computer system on which you ran your evolutionary algorithm. Include the processor or CPU (including the number of cores and clock speed), the amount of main memory, and the version of Python 3.

4.2 tour50.csv

Run your algorithm on this benchmark problem (with the 5 minute time limit from the Reporter). **Include a typical convergence graph, by plotting the mean and best objective values in function of the time** (for example based on the output of the Reporter class).

What is the best tour length you found? What is the corresponding sequence of cities?

Interpret your results. How do you rate the performance of your algorithm (time, memory, speed of convergence, diversity of population, quality of the best solution, etc)? Is your solution close to the optimal one?

Solve this problem 500 times and record the results. Make a histogram of the final mean fitnessess and the final best fitnesses of the 500 runs. Comment on this figure: is there a lot of variability in the results, what are the means and the standard deviations?

4.3 tour250.csv

Run your algorithm on this benchmark problem (with the 5 minute time limit from the Reporter). **Include a typical convergence graph, by plotting the mean and best objective values in function of the time** (for example based on the output of the Reporter class).

What is the best tour length you found in each case?

Interpret your results. How do you rate the performance of your algorithm (time, memory, speed of convergence, diversity of population, quality of the best solution, etc)? Is your solution close to the optimal one?

4.4   tour1000.csv

Answer the same questions as in **??**.

5   Critical reflection   (target: 1.5 **page**)

Goal: Based on this section, we will evaluate your understanding and insight into the main strengths and weaknesses of evolutionary algorithms, as well as your insights into responsible genAI usage.

5.1   Strengths and weaknesses

What are the three main strengths of evolutionary algorithms in your experience?

1.
2.
3.

What are the three main weak points of evolutionary algorithms in your experience?

1.
2.
3.

Describe the main lessons learned from this project. Do you believe evolutionary algorithms are appropriate for the problem studied in this project? Why (not)? What surprised you and why? What did you learn from this project?

5.2   GenAI peer reviewing critical reflection

What are the strengths and weaknesses of the AI-generated peer review compared to the human-written one? Did Copilot identify issues the humans missed, or vice versa? Why do you think this happened? Was Copilot's review at all useful? Was it specific and concrete or rather vague and general? What is your view towards reviewing reports using generative AI? Would you want to be reviewed by an AI system or by a human? Why? Did writing your own review help you understand the other group's algorithm better than reading the AI's version? How might relying too much on AI affect your ability to critically evaluate scientific work in the future? In what ways can AI be a helpful tool in peer review? In what ways can it be harmful?

5.3   GenAI critical reflection

In what ways did generative AI help you during the project? In what ways did it hinder your learning or understanding? Briefly describe for which activities you believe genAI added value:

1.
2.
3.

For which activities did you not observe any or sufficient added value:

1.
2.
3.

Would your project solution have been substantially different without relying on genAI? In what aspects? Why? How much time did you gain or lose by relying on genAI? If you did not substantially employ genAI: motivate why did you not use it.

6 Other comments (does not count toward page limit)

In case there is something important to discuss that is not covered by the previous sections, you can do it here.

7 Generative AI compliance form (does not count toward page limit)

For **each instance** of a genAI activity that required reporting (generating ideas and code), fill out the following information. You do not need to provide this information for the unconditionally approved uses of genAI.

7.1 Example activity

Briefly describe the activity for which you employed GenAI. For example: "Code skeleton generation for a recombination operator of permutations." Give the subsection an informative title.

Generative AI model: Identify the GenAI model used for this activity. For example, Microsoft Copilot, OpenAI ChatGPT v4, Midjourney, etc.

Motivation: Describe the intention and reason for employing generative AI for this activity.

Methodology: List the exact prompts you used to generate the content. You do not need to include the model's response.

Postprocessing: Describe which postprocessing you applied to the generated content. For example, "the output was used verbatim," "the output contained basic programming errors (undefined variables, undefined functions), which I corrected by hand," "I rewrote/removed/expanded one/a few/some/most/all sentences to [purpose]," "The generated content contained interesting ideas for a recombination operator, suggesting in particular to [do it in such and such way]. However, this idea would have been computationally very expensive, so I simplified [this and that], using [advanced data structure] to obtain an efficient variant of the generated idea."

Reflection: Critically reflect on the output generated and describe what you learned and in which aspects the result was useful and in which aspects it was not useful.

7.2 Another example activity

Generative AI model:

Motivation:

Methodology:

Postprocessing:

Reflection: