Genetic Algorithms and Evolutionary Computing
Template for project report

Please use this template to structure your report to facilitate the reading and evaluation. Do not repeat information from the textbook (e.g. if you use a crossover operator described in the book: don't copy the explanation or algorithm, but refer to the relevant page). The report should be ≈ 10 pages long; if useful you can add extra details in appendices (not included in the ≈ 10 pages).

1 Existing genetic algorithm

Perform *a limited set* of experiments by varying the parameters of the existing genetic algorithm (population size, probabilities, . . .) and evaluate the performance (quality of the solutions).

1. Data set(s) used & explain why you selected these data set(s)
2. Parameters considered and intervals/values
3. Performance criteria used
4. Test results
5. Discussion of test results

2 Stopping criterion

Implement a stopping criterion that avoids that rather useless iterations (generations) are computed.

1. Stopping criterion & explain why you selected this criterion
2. Test results (incl. performance criteria and parameter settings)
3. Discussion of test results

3 Other representation and appropriate operators (main task)

Implement and use another representation and appropriate crossover and mutation operators. Perform some parameter tuning to identify proper combinations of the parameters.

1. Representation
2. Crossover operators & explain why you selected the operators
3. Mutation operators & explain why you selected the operators
4. Parameter tuning: parameters considered and intervals/values
5. Data set(s) used & explain why you selected these data set(s)
6. Test results (incl. performance criteria and parameter settings)
7. Discussion of test results

4 Local optimisation

Test to which extent a local optimisation heuristic can improve the result.

1. Local optimisation heuristic & explain why you selected this heuristic
2. Test results (incl. performance criteria and parameter settings)
3. Discussion of test results

5 Benchmark problems

Test the performance of your algorithm using *some* benchmark problems (available on Toledo) and critically evaluate the achieved performance.

Keep in mind that for a large number of cities the search space is extremely large! If your algorithm doesn't perform well for a rather small number of cities, it doesn't make sense to use it for a benchmark problem with a large number of cities ...

Note: For most of the benchmark problems the length of the optimal tour is known. However, the Matlab template program scales the data. Therefore this scaling must be switched off to be able to compare your result with the optimal tour length.

1. List of benchmark problems
2. Test results (incl. performance criteria and parameter settings)
3. Discussion of test results

6 Other task(s)

You should select *at least one* task from the list below:

1. Implement and use two other parent selection methods, i.e. fitness proportional selection and tournament selection. Compare the results with those obtained using the default rank-based selection.
2. Implement one survivor selection strategy (besides the already implemented elitism). Perform experiments and evaluate the results.
3. Implement and use one of the techniques aimed at preserving population diversity (e.g. subpopulations/islands, crowding, . . .). Perform experiments and evaluate the results.
4. Incorporate an adaptive or self-adaptive parameter control strategy (e.g. parameters that depend on the state of the population, parameters that co-evolve with the population, . . .). Perform experiments and evaluate the results.

For each task:

1. Description of implementation
2. Description of the experiments
3. Test results
4. Discussion of test results

Time spent on the project

1. For each student of the team: estimate how many hours spent on the project (NOT including studying textbook and other reading material).

   (a)

   (b)

2. Briefly discuss how the work was distributed among the team members.