Genetic Algorithms and Evolutionary Computing Project

2015-2016

Introduction

The aim of the project is to implement a Genetic Algorithm to solve the Traveling Salesman Problem using the Genetic Algorithms Matlab Toolbox, to experiment with various representations, genetic operators and parameters, and to report on the results.

We prepared a template Matlab program for the TSP, using the Genetic Algorithms Matlab Toolbox and some extra procedures (Matlab .m files) for:

- input and visualization,
- representation (or coding) of a 'tour' using the adjacency representation,
- initialization of a population,
- recombination (or cross-over) using the alternating edge strategy,
- swap mutation operator
- conversion between adjacency representation and path representation.

Relevant procedures in the GA Toolbox are: `ranking.m, select.m, recombin.m, reins.m`. You can find more information about these functions in the GA Toolbox tutorial (available on Toledo). Relevant procedures in the template program are:

- `tspgui.m` is the main user interface script (use this function to start the program). Use this function as a quick demonstration of the available functions.
- `run_ga` is the main template program. You can use this file as basis for the project.
- `tspfun.m` is the fitness function for adjacency representation. If you decide to use path representation, you have to adapt this function
- `xalt_edges.m` which calls low-level recombinator function `cross_alternate_edges.m` (alternating edge cross-over)
- `mutateTSP.m` which calls low-level mutator function `reciprocal_exchange.m` or `inversion.m`
- `visualizeTSP.m` (visualization of a TSP tour)
- `adj2path.m, path2adj.m` (conversion between representations)

Tasks

1. On Toledo, you can find the GA Toolbox, the template program and tutorials about Matlab. Test the Matlab program to solve a TSP.

2. Perform limited experiments with the various parameters of the existing GA TSP (population size, probabilities, . . .) and evaluate the performance.

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

4. Test the performance of your algorithm using benchmark problems (available on Toledo). Indicate which ones of your modifications attribute to the performance gains.

5. Write a short report (≈ 10 pages, appendices and code not included), discussing your implementation and explaining your results. Include your code in the appendix.

Optional tasks

You can select one optional task from the list below:

1. Implement and use one of the techniques aimed at preserving population diversity (e.g. subpopulations/islands, crowding, . . .)

2. Implement and use two parent selection methods: proportional and tournament, and modify the default ranking selection behaviour. Compare the results with those obtained using default ranking method.

3. Implement one survivor selection strategy on top of the existing elitism. Perform experiments and evaluate the results against the already implemented elitism-based strategy.

4. Incorporate a strategy of adaptive or self-adaptive parameter control (e.g. parameters that depend on the state of the population, parameters that co-evolve with the population, . . .)

Experiment with various parameters that control these methods. Evaluate if there are differences in either performance or quality of solution.

Practical arrangements

- This project should be made in groups of 2 students. If this is not possible, please contact Prof. D. Roose asap. The estimated workload per student is 20 hours.

- The report (as a PDF file) and the source code (as a ZIP archive) have to be uploaded on Toledo: "Toledo/Assignments/Upload of the project report". Please name both the report and the archive file according to the pattern: *LastName1stAuthor-LastName2ndAuthor*. The hard copy of the report must be handed in by January 11th 2015 1 pm (Brussels local time) in the postbox near the secretariat of the Department of Computer Science (keep a copy of the report). The report will be briefly discussed during the examination.

Installing and using the Toolbox

The Genetic Algorithm Toolbox (`Toolbox.zip`) can be downloaded from Toledo. It is a slightly modified version of the original toolbox, which was developed at the University of Sheffield, UK.

The template Matlab programs for the TSP problem: `tsp.zip` (on Toledo).

Extract both files (tsp and toolbox) in a directory `../ga` and start Matlab. In Matlab click File... Set Path... Add with Subfolders. And then choose the directory `../ga`. Now Matlab can find both the toolbox and the template.

To change Matlab working directory to the location where you extracted the `tsp.zip` archive, double tap on that directory in the navigation panel. Start the algorithm by typing `tspgui` on the command line. The program initializes with 16 random cities. You can vary different parameters of the genetic algorithm using the sliders. Click 'input' to input a new set of cities. Click 'RUN' to run the genetic algorithm.

The user interface shows three different figures:

- Figure 1 (top left) shows the tour-length of the best individual at each generation.
- Figure 2 (top right) shows the evolution of the tour lengths (maximum, mean and minimum length at each generation).
- Figure 3 (bottom left) shows the histogram of tour length in the population. All the individuals in the population are put in bins with respect to their tour length: (x-axis: distance (tour length), y-axis: number of individuals). Figures 2 and 3 change at each generation step.

For your experiments you may find it useful to use the same placing for your cities several times. You can do this by changing the upper part of `tspgui` (where x and y are created) by e.g. (for NVAR=8) x=[0.1 0.2 0.3 0.4 0.5 0.2 0.3 0.4]'; y=[0.1 0.1 0.1 0.2 0.5 0.2 0.3 0.4]';

If you are unfamiliar with Matlab, there is an extensive Matlab Getting Started Tutorial available (on-line and printer-ready):

- `http://www.mathworks.com/access/helpdesk/help/techdoc/learn_matlab/bqr_2pl.html`
- `http://www.mathworks.com/access/helpdesk/help/pdf_doc/matlab/getstart.pdf`

More documentation about Matlab can be found on Mathworks Documentation Website:
`http://www.mathworks.com/access/helpdesk/help/techdoc/matlab.shtml`.