Assignment 2

COMP 9058 - Metaheuristic Optimisation

November 23, 2020

Due date:
Assignment should be submitted to Canvas in a single pdf by **Friday, Dec 18th, 23:59**. As per CIT regulations, submitting within 7 days of the deadline will result in a 10% penalty, between 7 and 14 days late will result in a 20% penalty, and later than 14 days after the due date will result in a 100% penalty applied.
The pdf MUST be named *Lastname_Firstname_StudentNumber_MH2.pdf*. Similarly the folder containing the code to be zipped MUST be named *Lastname_Firstname_StudentNumber_MH2_code*.
You need to write a report with a comprehensive description of the experiments and an extensive evaluation (and analysis) of the results. Use tables and figures where appropriate. As always, the random number generator must be seeded to your student number for your experiments.

1 Assignment Description

1.1 TSP - [70 Marks]

The basic 2-opt algorithm is outlined in algorithm 1 below. For this part of the assignment, you must implement and test the basic algorithm and two variants. The first variant is similar to N-Queens, in that rather than search all edges we choose an edge at random and we search all possible edges for swapping that edge with. In this scenario, we don't know if we are in a local optima so you must include a limit on number of iterations or number of non-improving iterations. The second variant is to change best improvement for first improvement, as soon as an improving move is found it is made.
You must evaluate your implementation of these three methods on the three TSP instances you were given in the first assignment (based on your surname) and discuss the results. You should also consider how efficiently your algorithm works when implementing.

Algorithm 1: 2-opt local search

```
Result: 2-optimal tour
T = InitializeTour();
notOpt = TRUE;
while notOpt do
    notOpt = FALSE;
    i = 0 ;
    currBest = cost(T) ;
    while i < n − 2 do
        j = i + 2 ;
        while j < n do
            D =SwapCost(i + 1, j − 1, T);
            if D < currBest then
                bestMove = [i, j] ;
                currBest = D ;
                notOpt = TRUE;
            end
            j + + ;
        end
        i + + ;
    end
    if notOpt then
        T = UpdateTour(T, bestMove) ;
    end
end
```

1.2 Runtime Distribution - [30 Marks]

In the attached you will find code for MaxMin hillclimbing for NQueens as discussed in the lecture. You must implement a variant that does not allow sideways moves. Analyse and compare the RTDs (with at least 100 executions) of these two algorithms. The value of n for your experiments will be the last two digits of your student id + 100. You should test different iteration limits and your choices for both.

2 Submission, marking and academic integrity

2.1 Submission:

This assignment is due on Friday, Dec 18th, 2020. You must submit the pdf separately via the Turnitin assignment submission: You must submit the following files (in a single zip file) via the code submission:

- All source code.
- A Readme file, which briefly describes all submitted files. In the Readme file, you should also provide information about compiling environment, compiling steps, execution instructions, etc.

Your pdf **MUST** be named Lastname_Firstname_StudentNumber_MH2.pdf (e.g. Grimes_Diarmuid_R001234567_MH2.pdf). Similarly the folder containing the code to be zipped MUST be named *Lastname_Firstname_StudentNumber_MH2_code*.

2.2 Rubics:

Coding

	The algorithm is logically well designed and efficient without inappropriate design choices (e.g., unnecessary loops). Code is well commented.	The algorithm always works properly and meets the specification. Code is clean, understandable and well organized.	The algorithm works properly in limited cases.	The algorithm is incorrectly implemented.
TSP x 3	(12-15 Marks)	(7-11 Marks)	(4-6 Marks)	(0-3 Marks)
N-Queens	(7-10 Marks)	(5-6 Marks)	(2-4 Marks)	(0-2 Marks)

Evaluation

	Excellent presentation, depth and insight analysis of the empirical results.	Good presentation of the results (e.g., describing the results with well structured tables).	Incomplete and/or unclear presentation of the results.	The results are inconsistent with the logic of the configuration/ operators.
TSP	(19-25 Marks)	(13-18 Marks)	(6-12 Marks)	(0-5 Marks)
N-Queens	(16-20 Marks)	(11-15 Marks)	(6-12 Marks)	(0-5 Marks)

2.3 Academic Integrity

This is an individual assignment. The work you submit must be your own. In no way, shape or form should you submit work as if it were your own when some or all of it is not. Any online source that is used must be cited. If you are

unsure on whether something should be cited, general rule of thumb is to err on the side of caution and include the citation. You can also ask me via email
Collusion: Given how much freedom there is in the assignment, everybody's work will be different. It will be obvious if there is collusion. All parties to collusion will be penalized.
Deliberate plagiarism: You must not plagiarise the programs, results, writings or other efforts of another student or any other third-party. Plagiarism will meet with severe penalties, which can include exclusion from the Institute.
Your report will be checked for signs of collusion, plagiarism, falsification and fabrication. You may be called to discuss your submission and implementation with me and this will inform the grading, any penalties and any disciplinary actions.