Simulation Course: Lab Exercise 3

Simulated Annealing (SA), Tabu Search (TS), and Genetic Algorithms (GA)

Exercise 1: This exercise aims at improving your hands-on skills of working with metaheuristics. First, download the zip file LabExercise3.zip from course homepage, unzip this file into your Matlab home directory. You will find three sub-folders containing three different implementations of Traveling Salesman Problem (TSP). In the following, you will find detail information about the source codes and data files:

- The subfolder SA_TSP contains the source code and the data files for simulated annealing. You have two options to run the program: command line and GUI. To run in command line, type *main_sa*, and for GUI version type *stsp*. To configure SA parameters, open the *main_sa.m* in the editor and change the parameters. The GUI version also allows the configuration of parameters. In both cases, you can choose which data file to use for your simulation. The data files are located in the same subfolder. They differ in the number of cities. These files contain the x-y coordinates of the cities, and the distance values are calculated by the program.
- The subfolder TS_TSP contains the source code and the data files for tabu search. The only option for running this program is via command line. Type *main_TS* to run the program. To change the parameters, open the *main_TS.m* in the editor, and change the parameters.
- The subfolder GA_TSP contains the source code and the data files for genetic algorithm. The only option for running this program is via command line. Type *main_ga* to run the program. To change the parameters, open the *main_ga.m* in the editor, and change the parameters. The actual implementing of the GA operations are in *tsp_ga.m.*

Our main goal is to learn how to experiment with individual programs by changing the parameters, and also compare the performance of the three programs. Some particular activities you should do are as follows: 40 082 for SA, 38 071 for TS and 34 331 for GA.

a. With the default data file and parameters (already been set in the script files), run each program separately and compare the final objective values. Which program does show a better performance? Try other source files (that are listed in the scripts, but you should uncomment them). Make sure the same file is chosen for all programs. Do this for at least three different data files and record your observations. "loadbays29()": 9 294 for SA, 9 791 for TS and 9 074 for GA. "loadeil101();": 1707 for SA, 731 for TS and 670 for GA.

"loadeil535()": 19 943 for SA, 2 500 for TS and 3 971 for GA.

b. Run SA program several times with the same parameters. Does the result change from one run to another? Why does or doesn't it change? Repeat this for the TA and GA programs as well, and report your observations.

c. Focus on each program, and investigate how the change of parameters affects the result. For SA, the parameters of interests are number of iterations, cooling rate, initial temperature and the number of iterations per temperature. In the TS, the parameters of

interests are tabu tenure and the number of iterations. In GA, the parameters of interests are the population size and the number of iterations.

d. Now turn your focus to the GA program only. The current code does not implement cross over operator. Try to modify *tsp_ga.m* by adding a cross over operator of your choice. You can use the one mentioned in Lecture H3 for the TSP.

Exercise 2: Implement a GA program for the following maximization problem:

$$\max_x f(x) = x\sin(10\pi x) + 1.0, \quad -1 \le x \le 2$$

with population size 10, cross over probability 0.25, and mutation probability 0.04.

You have two options to implement this GA: either by completely implementing the scripts on your own, or by using an embedded Matlab GUI for GA programming. If you choose the first option, follow this link to understand how to implement the scripts.

http://se.mathworks.com/help/gads/examples/constrained-minimization-using-the-genetic-algorithm.html

If you choose the second option, type *gatool* (or *optimtool* and then choose ga from the drop-down list) in the command line, and follow the instructions in the GUI (see this link for a quick introduction: https://www.youtube.com/watch?v=iAbR34PnGms).