# CT5141 Optimisation Assignment 2

James McDermott

**Topic** Facility location with real-valued metaheuristic optimisation. How to run experiments.

**Deadline** Midnight Wednesday 17 November (Week 11).

**Groups** Students may work in groups of 2, or solo. A student may not work with another student they have worked with on any other assignment in this module or other modules on this programme. In a group of 2, both students should submit and submissions will be assumed to be identical.

**Problem** The **facility location** problem is about finding the best locations for some facilities. In our problem, we have a list of cities, and we are going to build some supermarkets to serve the customers in these cities. The supermarkets could be anywhere, not just at the cities, i.e. a supermarket could be near a city or between two cities. There is no population outside the cities. Supermarkets have no limit on capacity. We want to locate the supermarkets so as to minimise the total travel time for all customers. The objective function is formulated in `supermarket.ipynb` with data in `supermarket_cities_locations.csv` and `supermarket_cities_populations.csv`. A plotting function is included. We will implement a Random Search approach to this problem, a Particle Swarm Optimisation approach, and a CMA approach.

**Requirements**

1. For CMA, use `pip install cma`. For RS and PSO you may use your own code, code given in CT5141, and/or code from the internet (with citation). You may import any module in the standard library or installable using `pip` or `conda`. Any `pip` or `conda` install commands should be documented in the code. No unused code should be included in the submission.
2. For the PSO hyperparameters `popsize`, $w$, $\phi_g$, $\phi_p$ (also known as $c_g$, $c_p$), choose at least two possible values for each hyperparameter and carry out a factorial-design experiment (i.e., all possible combinations) to find the best combination for this problem. No need to experiment with hyperparameters for CMA and RS. Include comments in code to explain the choice of hyperparameter values.
3. Use 5 runs for each algorithm/configuration, with random seeds $(0, \ldots, 4)$ (`random.seed` and/or `np.random.seed` and/or `cma seed` parameter). However, for CMA use $(1, \ldots 5)$ as here 0 means "no random seed". Use a fitness evaluation budget of 20,000 for each run. Record the mean elapsed time over the 5 runs by calling `time.time()` before and after. Note: on my laptop, the total elapsed time for an entire experiment (RS, CMA, PSO with 2 values for each PSO hyperparameter, 5 repetitions) was 120s.
4. Aim to automate the experiment so that if you were asked to re-run it, you could do so with a single function call or Python command, i.e. you do not need to run some parts, then edit the code, then run some other parts.
5. Include a table of results in the following format. Numbers in the table should not include meaningless digits, e.g. `72834.238734` could be shortened to `72834`. Suggestion (not mandatory): if you store results in a Pandas DataFrame `df` you can print them out nicely either in a notebook using `print(df)` or in Python using `df.to_markdown()` (requires `conda install tabulate`).

| Algorithm | `popsize` | $w$ | $\phi_g$ | $\phi_p$ | min | mean | stddev | time |
|---|---|---|---|---|---|---|---|---|
| RS | na | na | na | na | ... | ... | ... | ... |
| CMA | na | na | na | na | ... | ... | ... | ... |
| PSO | ... | ... | ... | ... | ... | ... | ... | ... |
| PSO | ... | ... | ... | ... | ... | ... | ... | ... |
| (etc) | ... | ... | ... | ... | ... | ... | ... | ... |
| PSO | ... | ... | ... | ... | ... | ... | ... | ... |

6. Plot the best solution ever found. Write a short paragraph (max 100 words) to state your findings. Compare algorithms and PSO hyperparameters. Refer to min, mean, stddev, and time.

**Submission** Submit a single `.ipynb` file containing code, results, and conclusions. Include student name(s) and ID(s) at the top of the file.

**Grading** This is worth 10% of the module. Grading is weighted as follows:

- 20% Experimental design
- 40% Implementation including automation
- 20% Table and plot
- 20% Conclusions

Possible penalties: incorrect submission format and other deviations from the spec.

**Policy** Students are reminded of the University's policy on plagiarism. Students may discuss the assignment with other students but must not look at other students' work (including from previous years), or allow others to look at theirs. Any online sources used must be cited with URL and date of access. Materials from CT5141 need not be cited. By making a submission you declare that you have abided by these conditions. Suspected infringements will be investigated and may be referred to NUI Galway authorities.