

Prof. Luca Maria Gambardella  
Faculty of Informatics  
Università della Svizzera italiana

Dear Prof. Gambardella and T.A. Mele,

I proudly present to you the final submission for this years AI Cup. Here you can find the final score sheet and necessary files to reproduce the results.

For my solution I used a meta-heuristic ant colony optimization algorithm to find the shortest path of the proposed problems. To push my system further, I implemented two local search algorithms, being 2-optimization and 3-optimization.

In order to operate on large matrix-like data structures, I implemented my solution using a linear algebra library called Eigen for c++. This allowed me to have an ant colony solution that runs significantly faster. To further optimize my code, I executed the GNU c++ compiler with a -Ofast flag in order to allow the compiler to pre-compute a significant amount of the calculations needed to run the colony.

Considering that the ant colony optimization has many parameters in order to produce best results, I decided to produce them using an automated system, rather than simply guessing them. That is why initially, I created a test case parameters, and iteratively improved on them by changing the parameter slightly and comparing the results. In case the new parameter affected the performance in a positive way, I would proceed with the new parameter, otherwise I would reject it. Every run was saved to a problem specific csv file to keep a record of the runs, and can be examined in the results directory.

There were several c++ files produced throughout the competition. In order to keep track of what the program is intended to produce and in what manner, the following naming convention was used: [meta-heuristic algorithm]\_[local search]\_[with count (to return more time-related data)].cpp.

Some programs are:

- aco.cpp
- aco\_2opt.cpp
- aco\_3opt.cpp
- aco\_2opt\_wc.cpp
- aco\_2opt\_3opt\_wc.cpp

In order to facilitate all the mentioned functionalities, I was using a trainer.ipynb notebook to create python blocks and documentation related to individual tasks, such as parameter tuning or execution. In order to reproduce the results please use the environment.yml file when creating a conda environment.

The final score achieved was: 1%

Mak Fazlic

---