Plan

Basic experiments:

- Implement M−T−Z or D−L model for the ATSP
- Implement G−G or G−G m. model for the ATSP
- Implement the D−F−J cut separation routine than identifies cuts violated by solution x^*_{ij}
- Strengthen the M−T−Z (D−L) or G−G (G−G m.) formulations with all violated D−F−J inequalities at the root node, i.e., separate and add violated inequalities to the LP relaxation of the formulation until no violated inequality exists

In the end you should obtain something that is called a cut-and-branch algorithm to the problem using a compact formulation of the problem.

Next step: add cuts at the fractional nodes to the D−F−J formulation

- Take the D−F−J working template code that adds cuts only at integer solution (incumbent) nodes
- Use the implemented separation routine to add violated cuts at the fractional nodes of the BnB tree
- Experiment with frequency of calling the fractional callback. Possible alternatives are call it for the first N nodes in the tree, every kth node in the tree, etc.

Note: you can not disable the incumbent callback when the problem formulation is incomplete!

You obtain a branch-and-cut algorithm to the problem formulated using the formulation with an exponential set of constraints.

Final step: branch-and-cut algorithms for compact formulations

- Take the M−T−Z or D−L model for the ATSP
- Take the G−G or G−G m. model for the ATSP
- Strengthen the M−T−Z (D−L) or G−G (G−G m.) formulations with all violated D−F−J inequalities at the root node, i.e., separate and add violated inequalities to the LP relaxation of the formulation until no violated inequality exists
- Add the fractional callback that identifies D−F−J cuts using existing cut separation routine during the branch-and-bound process
- Experiment with frequency of calling the fractional callback.

Note: you do not need the incumbent callback any longer when the problem formulation is complete!

You obtain a branch-and-cut algorithm to the problem formulated using a compact formulation.

Adapting the obtained codes to the case of STSP!

Practical Information

- You might want to form groups of 2
- One might focus on implementing a label setting formulation (D−L or M−T−Z) and the other might focus on implementing one of the one-commodity network flow models
- Work together on implementing the cut separation algorithm
- Verify each other's code and obtained solutions / bounds after cut separation algorithms terminate