

For progress, check 'meeting updates' folder.

Most of files are in markdown format. It is best to view in app obsidian or IDE like vscode with markdown plugin. The pdf version has the same content with markdown version.

There are two major methods to solve the problems. One is IP (integer programming), the other is local search.

Integer programming

To use IP solver to handle the problem, the first step is to formalize the problems into an LP (linear programming) problem. Then we could consider IP solver or modified IP solver.

For formalizations, see

formalizations

- |— formalization1_version1.0.pdf (segmentation + set cover)
- |— formalization2_version0.9.pdf (CVRP)
- |— formalization3_verion0.9.pdf (CVRP variant)

Solvers includes google's ortools, MiniZinc packages, gurobi

Note that IP method could be restricted by too many constraints and large inputs.

Local search

Our codebase is mainly based on local search. My project includes three parts:

1. analysis of our algorithm in production
2. case study of sub optimal solution produced by our algorithm
3. improve of our algorithm and design new algorithm.

— algorithm analysis drafts #1

- | — codes_trip_feasible().md
- | — compatible_group.md
- | — premerge.md
- | — pseudo code of exhaust booking.md
- | — remove old trips.md

| — algorithms #3

- | — 2 opt TSP.md
- | — Cutting Planes for MIP.pdf
- | — local search algorithm draft.md

| — case analysis #2

- | — blindspot of local search.jpg
- | — case 1 55 53 77.pdf
- | — case 1 analysis (53 55 77).md
- | — case 1 analysis (53 55 77).pdf