Airline Crew Pairing Reverse Optimization

Set Partitioning Workflow (SPP)

—----------

This repository implements a set partitioning approach to airline crew pairing selection. Given a large set of feasible crew pairings, the goal is to choose a minimum-cost subset such that each flight leg is covered exactly once.

The project is organized as a data-to-model pipeline:

- Parse and normalize pairing data
- Analyze and predict schedule costs. (Different cost modeling strategies implemented)
- Build the leg–pairing incidence structure
- Generate additional pairs from previous known solutions
- Solve the resulting Set Partitioning Problem (SPP) using integer programming

The modular structure makes it easy to swap in new pairing generators, cost models, or solvers without rewriting the entire pipeline.

Problem Formulation

$$x_j = \begin{cases} 1 & \text{if route } j \text{is selected,} \\ 0 & \text{otherwise,} \end{cases}$$

$$a_{ij} = \begin{cases} 1 & \text{if leg } i \text{is part of route } j, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$c_j = \text{cost of using route } j.$$

With these notations, the *equipment scheduling problem* is to

$$\begin{aligned} \text{minimize } & \sum_{j=1}^{n} c_j x_j \\ \text{subject to } & \sum_{j=1}^{n} a_{ij} x_j = 1 \qquad i = 1, 2, \ldots, m, \\ & x_j \in \{0, 1\} \qquad j = 1, 2, \ldots, n. \end{aligned}$$

Notes

This repository focuses on the pairing selection problem, not the generation of legal pairings themselves.
It is intended as a research-oriented and instructional implementation of SPP methods applied to airline crew scheduling.

Dependencies

Python 3.x
Julia
pulp (CBC solver)

Usage

git clone <repo_url>
cd <repo_folder>
pip install -r requirements.txt
python solver/phase_0_prep_the_data.py

Repository Structure

```
.
├── parsing/
│   └── phase_0_prep_the_data.py
├── preprocessing/
│   └── phase_1_a_schedule_analysis.py
    └──phase_1_b_cost_schedule_analysis.py
├── cost predictions/
│   └── phase_2_predict_cost_pipeline.py

├── create new pairings/
│   └── phase_3_set_generation_script.py
├── solver/
│   └── phase_3_set_generation_script.py

├── instances/
│   └── instance1/
│       ├── legs.csv
│       ├── pairings.csv
│       ├── incidence.csv
│       └── costs.csv
└── README.md
```

Data: G1422 dataset

Acknowledgements:

“*Airline crew scheduling: models, algorithms, and data sets*” Atoosa et. l