A Recovery Algorithm for a Disrupted Airline Schedule

Niklaus Eggenberg
Matteo Salani and Prof. M. Bierlaire

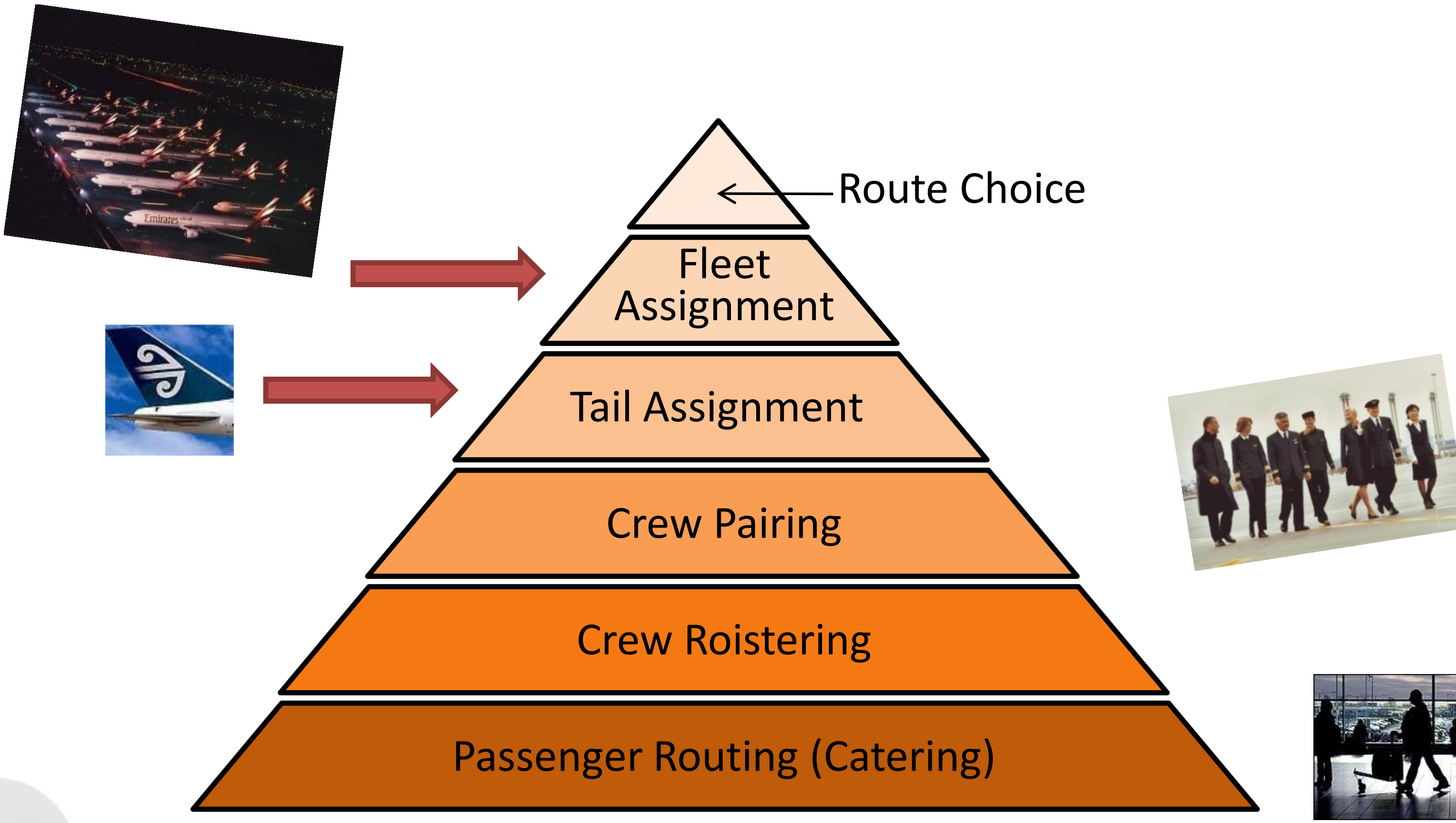
In collaboration with *APM Technologies*

TRANSP-OR
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

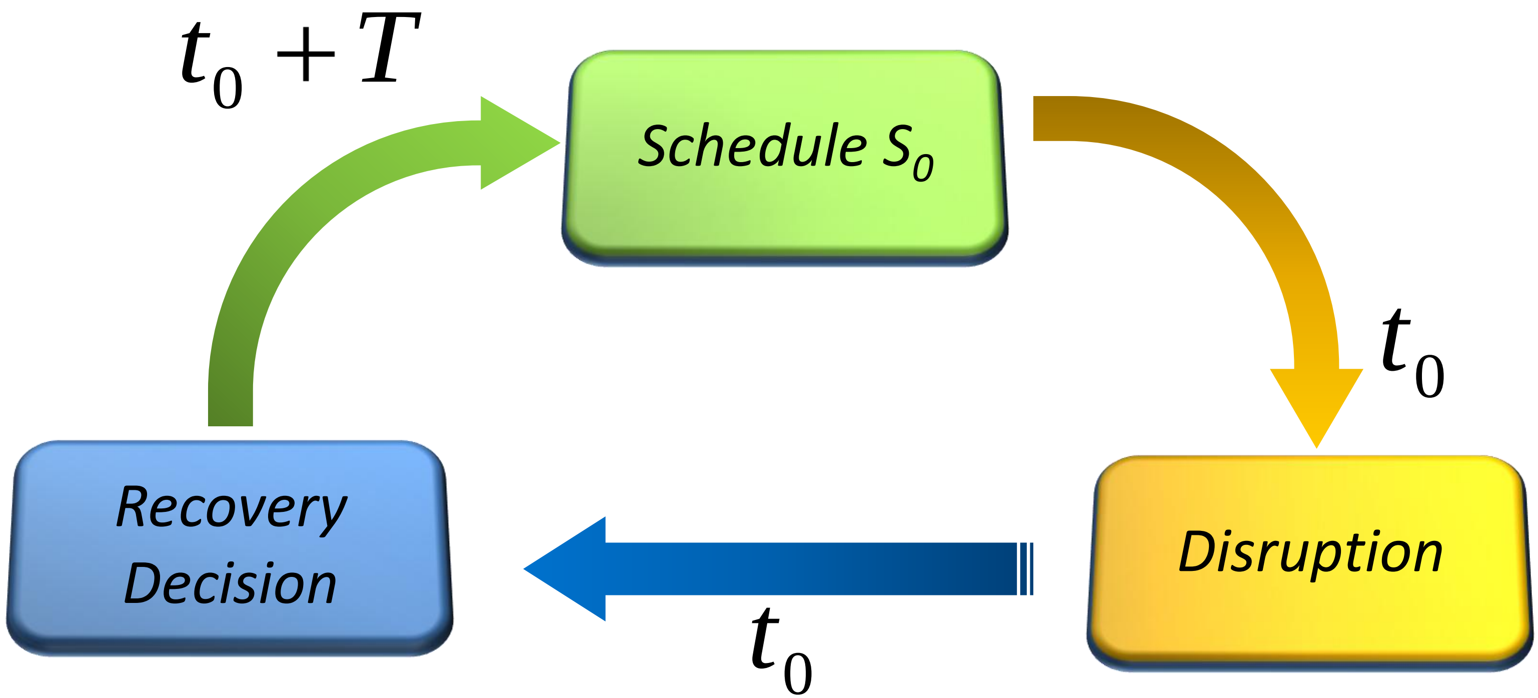
Index

- Airline Scheduling in general
- The Disrupted Schedule Recovery Problem (DSRP)
- The Column Generation (CG) approach
- Column Description
- The pricing algorithm with the Recovery Network
- Some preliminary results
- Future Work and Conclusions

Airline Scheduling Approach



Disrupted Schedule Recovery



Definitions

- ***Disruption***

  event making a schedule unrealizable

- ***Recovery***

  action to get back to initial schedule

- ***Recovery Period (T)***

  time needed to recover initial schedule

Definitions

- ***Recovery Plan***

  set of actions to recover disrupted schedule

- ***Recovery Scheme (r)***

  set of actions for a resource (plane)

Hypothesis

- consider only fleet and tail assignment
- no repositioning flights
- no early departure for flights
- work with universal time (UMT)
- initial state of resources are known
- no irregularity until end of recovery period
- maintenance forced by resource consumption

Column Generation

- column = recovery scheme (schedule for a plane)
- recovery scheme *r* = way to link Initial State to Final State with succession of flights and maintenances
- suppose set of all possible schemes *R* known
- find optimal combination of schemes

Master Problem (IMP)

$$\min \quad z_{MP} = \sum_{r \in R} c_r x_r + \sum_{f \in F} c_f y_f$$

$$\text{s.c.} \quad \sum_{r \in R} \boldsymbol{b}_r^f x_r + y_f = 1 \qquad \forall f \in F$$

$$\sum_{r \in R} \boldsymbol{b}_r^s x_r = 1 \qquad \forall s \in S$$

$$\sum_{r \in R} \boldsymbol{b}_r^p x_r \leq 1 \qquad \forall p \in P$$

$$x_r \in \{0,1\} \qquad \forall r \in R$$

$$y_f \in \{0,1\} \qquad \forall f \in F$$

What is a column ?

- vector $\boldsymbol{b}_r = \left(b_r^f, b_r^s, b_r^p \right)^T$

Where

- b_r^f = 1 if flight *f* is covered by column *r*
- b_r^s = 1 if final state *s* is covered by *r*
- b_r^p = 1 if column *r* is affected to plane *p*

Example

f_1 GVA to AMS

f_2 AMS to BCN

f_3 BCN to GVA

f_4 MIL to BUD

f_5 BUD to MIL

f_6 BCN to MIL

Example

- flights: $F = \{f_1, f_2, f_3, f_4, f_5, f_6\}$
- final states: $S = \{s^{GVA}, s^{MIL}\}$
- planes: $P = \{p_1, p_2\}$
- p_1 starts in *GVA*, p_2 starts in *MIL*

Column examples

$$\boldsymbol{b}_1 = (0,0,0,0,0,0,1,0,1,0)^T$$

$$\boldsymbol{b}_2 = (1,1,1,0,0,0,1,0,1,0)^T$$

$$\boldsymbol{b}_3 = (0,0,0,1,1,0,0,1,0,1)^T$$

Feasible Solution

Solving the Master Problem

I. Solve IMP with Branch and Bound

II. Solve linear relaxation LP at each node:

- Restrict LP to sub-set $R' \subseteq R$
- Solve RLP
- Find $\boldsymbol{b}_r \in R \setminus R'$ minimizing reduced cost
- Insert column if *r.c. < 0* and resolve RLP

The Pricing Problem

Find column $\boldsymbol{b}_r \in R \setminus R'$ minimizing reduced cost $\tilde{c}_r^p$

$$\min_{r \in R} \tilde{c}_r^p = c_p^r - \sum_{f \in F} \boldsymbol{b}_r^f \lambda_f - \sum_{s \in S} \boldsymbol{b}_r^s \eta_s - \boldsymbol{b}_r^p \mu_p \qquad \forall p \in P$$

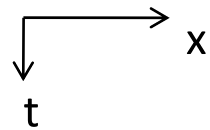
Recovery Network Model

↓

Solve Resource Constrained Elementary Shortest Path Problem (RCESPP)

The Recovery Network (Argüello et al. 97)

- Time-space network

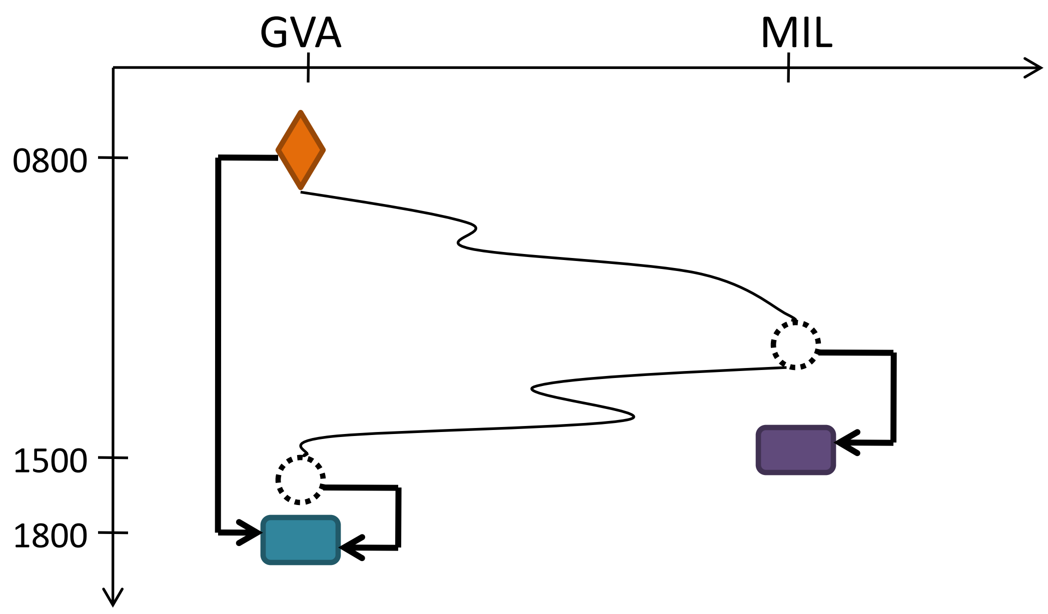


- One network for every plane
- Source node corresponding to initial state
- Sinks corresponding to expected final states
- 3 arc types (NEVER horizontal):
    1. Flight arcs
    2. Maintenance arcs
    3. Termination arcs (vertical)

Source and Sink Nodes

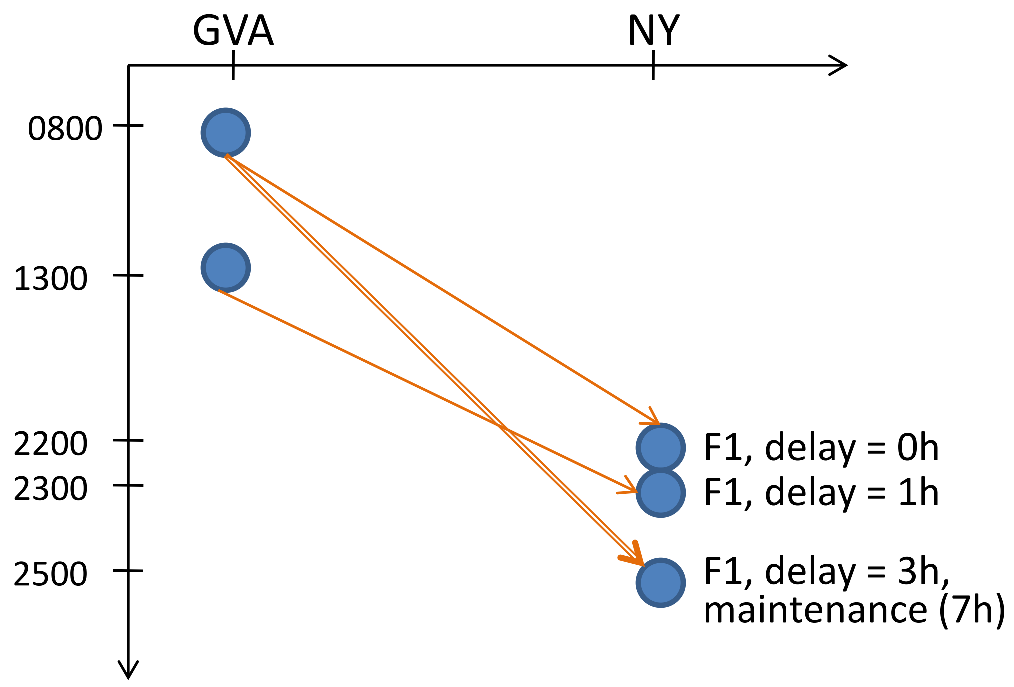
Plane p_1, initial state = [GVA, 0800]

Expected States : [GVA, 1800] and [MIL, 1500]



Flight and Maintenance Arcs

flight F1: GVA to NY at 1200



Arc Costs

- Flight arcs: $c = c^f - \lambda_f$
- Maintenance arcs: $c = c^f + c^M - \lambda_f$
- Termination arcs: $c = -\eta_s$

Recovery Network Properties

- No horizontal arcs
- No vertical arcs except termination arcs
- Node only at earliest availability time
- Grounding time included in arc length (3 types)
- Maintenances are integrated before flight if possible

Preliminary Results

- implementation using COIN-OR BCP
- solve three problems of various sizes:
  1. 48 flights, 9 airports, 3 planes
  2. 84 flights, 15 airports, 11 planes
  3. 36 flights, 17 airports, 10 planes
- solved 1. to optimality (root node)
- promising results for instances 2. and 3.

Future Work

- Work on implementation
- Test more real instances
- Explore more widely RCESPP and CG algorithms
- Compare solutions to real recovery decisions
- Include Algorithm in APM Framework

Conclusions

- Colum Generation to solve DSRP
- Adapted model to solve pricing problem
- Get quick solutions for decision aid
- Still need real-instance validation

THANKS for your attention!

Any Questions?