15-853:Algorithms in the Real World

Linear and Integer Programming IV
- Case Study: Airline Crew Scheduling

American Airlines

Problem: Schedule crew (pilots and flight attendants) on flight segments to minimize cost.
- over 25,000 pilots and flight attendants
- over $1.5 Billion/year in crew costs

Assumes the flight segments are already fixed.

Methods described today:
- 1970-1992: TRIP (Trip Reevaluation and Improvement Program). Local optimization given an initial guess
- 1992-present? Global optimization (approximate)

Algorithms in the Real World

Editor's note from a recent paper: "A crew resource planning manager at the subject company confirmed to me that the work described in this paper has been in regular operational use since its completion, that it has been used to support labor negotiations, and that while its benefits have not been quantified, the system is an improvement over the prior system and is working well. **For commercial reasons, the subject company wants to remain anonymous.**

Crew Pairings

Example: A 2 day **crew pairing** with DFW (Dallas-Fort Worth) as the base.

Duty period 1
- Sign in: 8:00
- DFW 9:00-10:00 AUS (segment 1)
- AUS 11:00-13:00 ORD (segment 2)
- ORD 14:00-15:00 SFO (segment 3)

Overlay in SFO

Duty period 2
- Sign in: 7:00
- SFO 8:00-9:00 LAX (segment 4)
- LAX 10:00-11:45 SAN (segment 5)
- SAN 13:00-19:30 DFW (segment 6)
- Sign out: 19:45

Properties of Pairings

National pairings typically last 2 or 3 days.

Crew work 4 or 5 pairings per month.

Collection of pairings in a month is a **Bidline**.
- Recent work has considered optimizing the bidlines. Today we will just discuss pairings.
- Crew "bid" on the bidlines (seniority based)

Constraints on Pairings

Union and Federal Aviation Agency (FAA) rules

Some **example constraints**
- 8 hours flying per duty period
- 12 hours total duty time
- Minimum layover time – depends on hours of flying in previous duty period
- Minimum time between flights in a duty period

Cost of Pairings

Cost can include both direct and indirect costs (e.g. employee satisfaction).

Example contributions to "cost".
- Total duty period time
- Time away from base (TAFB)
- Number and locations of overlays
- Number of time zone changes
- Cost of changing planes

Overall Goal

Cover all segments with a set of valid pairings that minimize costs.

Must also consider number of crew available at crew bases.

Crew pairings (as well as flight schedules) are generated on a monthly basis.

Problem is simplified since flights are pretty much the same every day. The monthly boundaries can cause some problems.

Possible Approach

Consider **all** valid pairings and generate cost for each.
Now solve as a **set covering** problem:

Given m sets and n items:

$$A_{ij} = \begin{cases} 1, & \text{if set } j \text{ includes item } i \\ 0, & \text{otherwise} \end{cases}$$

$$c_i = \text{cost of set i}$$

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

minimize: $c^T x$

subject to: $Ax \geq 1$, x binary

Problem: Billions of possible pairings

Example Formulation

Segments to be covered:

1. DFW 9-12 LGA
2. LGA 13-15 ORD
3. ORD 16-18 RDU
4. ORD 17-19 DFW
5. RDU 19-21 LGA
6. RDU 19-21 DFW
7. LGA 14-16 ORD
8. DFW 16-18 RDU

Pairings:

1. DFW 9-12 LGA 14-16 ORD 17-19 DFW (1,7,4)
2. LGA 13-15 ORD 16-18 RDU 19-21 LGA (2,3,5)
3. ORD 16-18 RDU 19-21 DFW 9-12 LGA 13-15 ORD (3,6,1,2)
4. DFW 16-18 RDU 19-21 DFW (8,6)
5. DFW 16-18 RDU 19-21 LGA 14-16 ORD 17-19 DFW (8,5,7,4)

Example Formulation (cont.)

$$A^T = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

$$c = [c_1 \quad c_2 \quad c_3 \quad c_4 \quad c_5]$$

Old System, TRIP (->1992)

1. Select an initial solution (set of pairings)
   Typically a modification from previous month
2. Repeat the following until no more improvements:
   - Select small set of pairings from current solution
     Typically 5-10 pairings from the same region
   - Generate all valid pairings that cover the same segments, and cost for each
     Typically a few thousand
   - Optimize over these pairings using the set-partitioning problem.

Advantage: Small subproblems

Problem: Only does local optimization

New System (?)

Anbil, Tanga, Jonhson, 1992:

1. Generate large "pool: of pairings
   About 6 million.
2. Solve LP approximation using specialized techniques
3. Use "branch-and-bound" for IP solution, with heuristic pruning

Each LP takes about an hour (possibly faster now)

Does not guarantee best solution because of the pruning step, but much better than TRIP.

Generating the Pool of Pairings

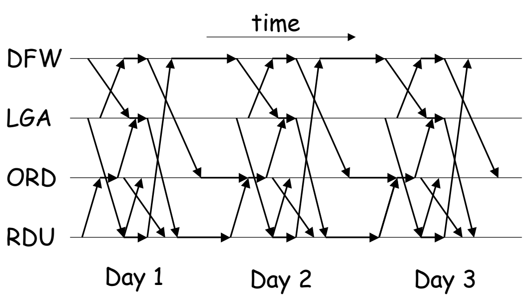
Disclaimer: This is speculative since the authors say very little about it.

Generating 6 million initial pairings out of billions of possible pairings

1. Generate graph
   - Vertex: time and airport
   - Edge: flight-segment, wait-time, or overlay
   - Edge weight: "excess cost" of edge
2. Find 6 million shortest valid paths (e.g.by union rules) in the graph that start and end at a crew base

This is a heuristic that prefers short TAFB.

Shortest Path Graph



Each edge is given a weight based on approximate cost (full cost not known without rest of pairing)

Solving the LP Approximation

1. Select a small set of m columns (pairings), Call this submatrix A. $m \approx 5000$
2. Repeat until optimal solution found
   - Optimize problem based on A
   - Use the basic variables to "price" the remaining variables and set A to the m "best (i.e. pick 5000 minimum reduced costs $r = c_b B^{-1} N - c_n$)

Using the LP for the IP

Algorithm:

- Solve the LP approximation across 6 million columns
- Select about 10K pairings with best reduced cost
- Repeat until all segments have a follow on:
  1. For all non-zero pairings, consider all adjacent segments (s_i,s_j) in the itineraries
  2. Add weights from the pairing that include them, and select maximum sum across all (s_i, s_j).
  3. Fix (s_i, s_j) and throw out all pairings that include (s_i,s_k), $k \neq j$
  4. Solve the LP again
  5. Add new columns from original 6 million if system becomes infeasible

Example: from before

$$A^T = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

segments	LP solution
1 7 4	1/2
2 3 5	1/2
3 6 1 2	1/2
8 6	1/2
8 5 7 4	1/2

Segment pairs	(1,7)	(1,2)	(2,3)	(3,5)	(7,4)	rest
Summed weights	1/2	1/2	1	1/2	1	1/2

We therefore fix (2,3) : LGA 13-15 ORD 16-18 RDU
and (7,4): LGA 14-16 ORD 17-19 DFW
We don't throw out any pairings since 2 and 7 are not followed by anything other than 3 and 4, respectively

Additional Constraints

Need to account for the number of crew available at each base

- Add constraints with maximum and minimum hours available from each base

Need to patch between months.

- Separately schedule first two days of each month with additional constraints put in from previous month.

Handling cancelled or delayed flights.

- Currently done by hand – every base has a small set of reserve cre.

Some Conclusions

- Use of special purpose techniques
- Mostly separates the optimization from the cost and constraints rules.
- Solves 6 million variable LP as a substep.
- It is hard to get specifics on money saved (initial papers were much more forthcoming)