A Delayed Column Generation Approach for Solving a Cargo Crew Scheduling Problem

Junhong Guo    Amy Cohn

Department of Industrial and Operations Engineering
University of Michigan, Ann Arbor, MI

Oct. 20, 2019
Outline

1. Introduction
2. Model and Solution Framework
3. Basic Case: Crew Pairings with No Breaks
4. Advanced Case: Crew Pairings That Allow Breaks
5. Generalization
6. Conclusion and Future Work
Cargo Aviation Industry

- Global air cargo traffic is forecast to grow a robust 4.2% per year over the next 20 years (Boeing 2017)
  - The revenue ton-kilometers (RKT) will more than double from 256 billion in 2017 to 584 billion in 2037
  - The number of freighter airplanes will grow by more than 70 percent in total
The cargo airline that we partnered with accepts requests for goods delivery, from one location to another, from customers including logistic companies, manufacturers, the military, and so on.

Requests are gathered and further partitioned into different planning horizons, typically a calendar month.

The cargo airline has a fleet of airplanes and a group of crews it can contract, and needs to determine and schedule all necessary tasks and activities accordingly so that the requests in each planning horizon will be delivered as planned.
Cargo Aviation Planning Procedure

- Schedule Design
- Fleet Assignment
- Aircraft Routing
- Cargo Routing
- Crew Scheduling
Cargo Aviation Planning Procedure

1. Schedule Design
2. Fleet Assignment
3. Aircraft Routing
4. Cargo Routing
5. Crew Scheduling
Crew Pairing

A sequence of flights that will be assigned to a single crew to carry out

- Specific requirements like labor regulations must be satisfied
Crew Pairing

A sequence of flights that will be assigned to a single crew to carry out
- Specific requirements like labor regulations must be satisfied

Traditional Passenger Aviation:

- Crew Scheduling Problem
- Crew Pairing Problem
- Crew Rostering Problem
Our Cargo Aviation Problem:

- The majority of flights are long-haul, international flights
- Lack repeating daily pattern

J. Guo, A. Cohn (UMICH)
AAS Presentation Competition
Oct. 20, 2019
Our Cargo Aviation Problem:

- Each crew pairing spans a much longer time, e.g. a half month
- The crew pairing will correspond to a complete crew schedule
- The crew are possibly away from home for the whole duration of the assigned pairing
- The crew often fly commercially from/to their base when starting/finishing the pairing

- The majority of flights are long-haul, international flights
- Lack repeating daily pattern
Introduction

Problem Statement

Requirements

- Basic “laws” of physics
- Regulatory policies
  - E.g. Each duty period cannot exceed 17 hours, and the crew must have a minimum 10-hour layover for rest before starting next duty period
- Corporate policies
  - E.g. The time span of the crew pairing must be at least 12 days
We have a different objective, unlike the traditional ones.
We have a **different objective**, unlike the traditional ones

- UNABLE to cover ALL flights in the planning horizon
  - The structure of the network - lacks opportunities for quick turns; includes many airports with a small number of associated flights
  - The requirements - avoid assigning short pairings to crews
We have a **different objective**, unlike the traditional ones

- UNABLE to cover ALL flights in the planning horizon
  - The structure of the network - lacks opportunities for quick turns; includes many airports with a small number of associated flights
  - The requirements - avoid assigning short pairings to crews
- The airline chooses to subcontract some of the scheduled flights
We have a **different objective**, unlike the traditional ones

- UNABLE to cover ALL flights in the planning horizon
  - The structure of the network - lacks opportunities for quick turns; includes many airports with a small number of associated flights
  - The requirements - avoid assigning short pairings to crews

- The airline chooses to subcontract some of the scheduled flights

**Objective**

Cover as many flights in the planning horizon as possible with valid crew pairings that satisfy all of the requirements
The airline targets to achieve 80% flight coverage

In practice, even the best solution is much lower than this rate
Introduction
Problem Statement

- The airline targets to achieve 80% flight coverage
- In practice, even the best solution is much lower than this rate

Remedy

Allow a “break” to take place in the “middle” of the crew pairing

- Two extra requirements must be satisfied to have a break:
  - The duration of the break must be at least 6 days
  - The break cannot take place before the crew completes the fourth flight, or after the seventh flight in the pairing
Introduction
Problem Statement

- The airline targets to achieve **80%** flight coverage
- In practice, even the best solution is much lower than this rate

Remedy

Allow a "break" to take place in the "middle" of the crew pairing

- Two extra requirements must be satisfied to have a break:
  - The duration of the break must be at least 6 days
  - The break cannot take place before the crew completes the fourth flight, or after the seventh flight in the pairing

Basic Case: **No Breaks**

Advanced Case: **Allow Breaks**
Model and Solution Framework
A Set Packing Problem

Formulation

\[
\begin{align*}
\text{min} & \quad \sum_{p \in P} -n_p \cdot x_p \\
\text{s.t.} & \quad \sum_{p \in P} a_{f,p} \cdot x_p \leq 1 \quad \forall f \in F \\
& \quad x_p \in \{0, 1\} \quad \forall p \in P
\end{align*}
\]

Sets and Parameters

\(F\) the set of flights
\(P\) the set of valid crew pairings
\(n_p\) the number of flights covered by pairing \(p\), for \(\forall p \in P\)
\(a_{f,p}\) 1 if flight \(f\) is covered by pairing \(p\); 0 otherwise, for \(\forall f \in F, \forall p \in P\)

Decision Variables

\(x_p\) Binary for \(\forall p \in P\). 1 if pairing \(p\) is assigned to a crew; 0 otherwise
First, solve the LP-relaxation to optimality, with the crew pairings iteratively incorporated on demand, driven by the dual values via a delayed column generation (DCG) framework (Lavoie et al. (1988), Anbil et al. (1998), Wei and Vaze (2018), ...)

Heuristically solve the original integrality-constrained set packing formulation with pairings limited to those generated during the DCG (Barnhart et al. (1994), Dunbar et al. (2012), ...)

First, solve the LP-relaxation to optimality, with the crew pairings iteratively incorporated on demand, driven by the dual values via a delayed column generation (DCG) framework (Lavoie et al. (1988), Anbil et al. (1998), Wei and Vaze (2018), ...)

Heuristically solve the original integrality-constrained set packing formulation with pairings limited to those generated during the DCG (Barnhart et al. (1994), Dunbar et al. (2012), ...).
Model and Solution Framework

Delayed Column Generation

Initialization → Solve the Restricted Master Problem → Add Them to the Master Problem → Generate the Corresponding Columns

Exist Crew Pairing(s) with Negative Reduced Cost? [YES] → DCG [NO] → Heuristically Solve the Set Packing Formulation
Generating Pairings with No Breaks via SPPRC

Previous Work

- Shortest Path Problem with Resource Constraints (SPPRC) was first introduced for solving a routing problem with time windows for bus transportation (Desrosiers et al. (1984)).

- It has since been generalized, and several variants have been proposed, to address a wide range of problems in transportation:
  - **Routing:** Desrochers and Soumis (1989), Dumas et al. (1991), Ioachim et al. (1998), Feillet et al. (2004), ...
  - **Crew Scheduling:** Vance et al. (1997), Gamache et al. (1999), Dunbar et al. (2012), Shao et al. (2015), ...
  - Etc.

- Irnich and Desaulniers (2005) provides a comprehensive instruction and survey on SPPRC.
Generating Pairings with No Breaks via SPPRC

Flight-based Network

Directed graph $G(V, E)$

$V$: Nodes in the network, consist of $F \cup \{s, t\}$

$E$: Arcs in the network. For $\forall f_1, f_2 \in F$, arc $(f_1, f_2)$ exists if and only if the basic "laws" of physics (i.e. Requirement 1) hold on this follow-up. In addition, for $\forall f \in F$, there are arcs $(s, f)$ and $(f, t)$

An $s - t$ path in the network corresponds to a potential crew pairing.
Every path $p$ in the network $G$ is tracked with a resource vector $T^p \in \mathbb{R}^k$:

- Each of the first $k - 1$ resources is defined to prevent the violation of one of the remaining requirements
  
  E.g. $r_a$: The amount of time the current duty period has spanned so far (Ensure no duty period exceeds the maximum 17-hour duty length)

  E.g. $r_b$: The remaining amount of total time required by the current crew pairing to fulfill the minimum requirement on the total time span (Ensure the length of the finalized crew pairing at the end is not shorter than 12 days)

- The last resource $r_k$ is defined for the calculation of the reduced cost of the current crew pairing
Let $\mathcal{P}$ be the set of feasible $s - t$ paths respecting all resource constraints in the network $G$.

The original pricing problem is then equivalent to solving the following formulation based on our SPPRC model:

$$\min_{p \in \mathcal{P}} T_k^p$$
Generating Pairings with No Breaks via SPPRC Model

- Let $\mathcal{P}$ be the set of feasible $s - t$ paths respecting all resource constraints in the network $G$.

The original pricing problem is then equivalent to solving the following formulation based on our SPPRC model:

$$\min_{p \in \mathcal{P}} T_k^p$$

- A **label-setting algorithm** is used to solve this formulation
  - Similar like Dijkstra’s algorithm, but in a higher dimension
  - Infeasible and inferior sub-paths are discarded during the path extension
### Generating Pairings with No Breaks via SPPRC

#### Computational Experiments

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Flights</th>
<th>#Valid Pairings</th>
<th>Enum. Time</th>
<th>#Flights Cov.</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>606</td>
<td>440,641</td>
<td>30min 34sec</td>
<td>332</td>
</tr>
<tr>
<td>No.2</td>
<td>541</td>
<td>329,145</td>
<td>26min 40sec</td>
<td>281</td>
</tr>
<tr>
<td>No.3</td>
<td>644</td>
<td>462,395</td>
<td>35min 52sec</td>
<td>334</td>
</tr>
</tbody>
</table>

J. Guo, A. Cohn (UMICH)
Computational Experiments

### Dataset: Flights

<table>
<thead>
<tr>
<th>No.</th>
<th>Flights</th>
<th>Valid Pairings</th>
<th>Enum. Time</th>
<th>Flights Cov.</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>606</td>
<td>440,641</td>
<td>30min 34sec</td>
<td>332</td>
</tr>
<tr>
<td>No.2</td>
<td>541</td>
<td>329,145</td>
<td>26min 40sec</td>
<td>281</td>
</tr>
<tr>
<td>No.3</td>
<td>644</td>
<td>462,395</td>
<td>35min 52sec</td>
<td>334</td>
</tr>
</tbody>
</table>

### Dataset: LP-obj

<table>
<thead>
<tr>
<th>No.</th>
<th>LP-obj</th>
<th>#Itr.</th>
<th>LP Time</th>
<th>Pairings Gen.</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>336.382</td>
<td>11</td>
<td>6min 13sec</td>
<td>22,052</td>
</tr>
<tr>
<td>No.2</td>
<td>284.447</td>
<td>9</td>
<td>3min 36sec</td>
<td>16,642</td>
</tr>
<tr>
<td>No.3</td>
<td>340.327</td>
<td>10</td>
<td>6min 24sec</td>
<td>23,736</td>
</tr>
</tbody>
</table>

### Dataset: IP-obj

<table>
<thead>
<tr>
<th>No.</th>
<th>IP-obj</th>
<th>IP Time</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>332</td>
<td>28sec</td>
<td>54.79%</td>
</tr>
<tr>
<td>No.2</td>
<td>281</td>
<td>24sec</td>
<td>51.94%</td>
</tr>
<tr>
<td>No.3</td>
<td>333</td>
<td>5min 20sec</td>
<td>51.71%</td>
</tr>
</tbody>
</table>
Generating Pairings That Allow Breaks
A Straight Extension

An additional set of "break arcs" $B$ are introduced into the flight-based network $G(V, \bar{E})$, where $\bar{E} = E \cup B$. The SPPRC model is updated accordingly, i.e., necessary resources are additionally incorporated to ensure the requirements introduced by the break feature are satisfied.
An additional set of “break arcs” $B$ are introduced into the flight-based network $G(V, \bar{E})$, where $\bar{E} = E \cup B$. 

![Diagram of flight-based network with break arcs](image-url)
An additional set of “break arcs” \( B \) are introduced into the flight-based network \( G(V, \bar{E}) \), where \( \bar{E} = E \cup B \).

The SPPRC model is updated accordingly, i.e. necessary resources are additionally incorporated to ensure the requirements introduced by the break feature are satisfied.
Computational Challenge

For the first iteration:

- Runtime: > 10 hours
- \#pairings found: > 2.6 million
Computational Challenge

For the first iteration:

- Runtime: > 10 hours
- #pairings found: > 2.6 million

<table>
<thead>
<tr>
<th>#Nodes</th>
<th>#Arcs (no breaks)</th>
<th>#Arcs (allow breaks)</th>
</tr>
</thead>
<tbody>
<tr>
<td>608</td>
<td>12,539</td>
<td>123,612</td>
</tr>
</tbody>
</table>
Generating Pairings That Allow Breaks
A Heuristic - Flight Partitioning Approach

1. Randomly Split the Set of Flights Evenly into Two Sets
   - Construct the Network with Break Arcs Based on the First Set
   - Construct the Network with Break Arcs Based on the Second Set

2. Solve the Corresponding SPPRC Model by the Label-Setting Algorithm
   - Add Pairings to the Master Problem; continue the DCG

3. Find Crew Pairing(s) with Negative Reduced Cost from either of Them?
   - YES
   - NO

   TERMINATE the DCG
Rather than dumping ALL break arcs into the original network, we only introduce a SUBSET of them each time (Barnhart et al. (1995)). Each selected arc should be beneficial. That is, crew pairing(s) containing the corresponding break which has a negative reduced cost should then be introduced.
Rather than dumping ALL break arcs into the original network, we only introduce a SUBSET of them each time (Barnhart et al. (1995))
Generating Pairings That Allow Breaks

An Exact Algorithm - Arc Selection Approach

- Rather than dumping ALL break arcs into the original network, we only introduce a SUBSET of them each time (Barnhart et al. (1995))

- Each selected arc should be beneficial. That is, crew pairing(s) containing the corresponding break which has a negative reduced cost should then be introduced.
Rather than dumping ALL break arcs into the original network, we only introduce a SUBSET of them each time (Barnhart et al. (1995)). Each selected arc should be beneficial. That is, crew pairing(s) containing the corresponding break which has a negative reduced cost should then be introduced.
Rather than dumping ALL break arcs into the original network, we only introduce a SUBSET of them each time (Barnhart et al. (1995)).

Each selected arc should be beneficial. That is, crew pairing(s) containing the corresponding break which has a negative reduced cost should then be introduced.
Generating Pairings That Allow Breaks
An Exact Algorithm - Arc Selection Approach

Forward

\[ f_i \]

\[ \text{...} \]

\[ t \]

Backward

\[ f_j \]
Generating Pairings That Allow Breaks
An Exact Algorithm - Arc Selection Approach

Do Forward

Loop through All Break Arcs \((f_i, f_j)\)

Do Backward

Loop through All Efficient Sub-Paths, Forward from \(s\) to \(f_i\)

Loop through All Efficient Sub-Paths, Backward from \(t\) to \(f_j\)

The Concatenation Corresponds to a Feasible Pairing with Negative Reduced Cost?

YES

Add \((f_i, f_j)\) to the Candidate Pool; Move to Next Break Arc

Solve the Extended SPPRC Model

Add Them to the Original Network

Randomly Select up to \(H\) Arcs from the Candidate Pool
The accumulation of resource consumption is independent augmentation, or independent augmentation with reset for all resources.

**Proposition**

A break arc is pruned out during the proposed arc selection procedure iff there does NOT exist any negative reduced cost crew pairing which contains a break corresponding to this arc.
The accumulation of resource consumption is independent augmentation, or independent augmentation with reset for all resources.

**Proposition**

A break arc is pruned out during the proposed arc selection procedure iff there does NOT exist any negative reduced cost crew pairing which contains a break corresponding to this arc.

- If NO pairing is found by solving the extended SPPRC model, then:
  - NO break arc is selected and added to the network
    - NO negative reduced cost pairing which contains a break exists
  - NO negative reduced cost pairing which does not contain a break exists
Generating Pairings That Allow Breaks
Computational Experiments

Basic Case: No Breaks

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Flights</th>
<th>#Arcs</th>
<th>#Valid Pairings</th>
<th>Enum. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>606</td>
<td>12,539</td>
<td>440,641</td>
<td>30min 34sec</td>
</tr>
<tr>
<td>No.2</td>
<td>541</td>
<td>10,113</td>
<td>329,145</td>
<td>26min 40sec</td>
</tr>
<tr>
<td>No.3</td>
<td>644</td>
<td>12,201</td>
<td>462,395</td>
<td>35min 52sec</td>
</tr>
</tbody>
</table>

Advanced Case: Allow Breaks

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Flights</th>
<th>#Arcs</th>
<th>#Valid Pairings</th>
<th>Enum. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>606</td>
<td>123,612</td>
<td>142,777,637</td>
<td>3day 02hr</td>
</tr>
<tr>
<td>No.2</td>
<td>541</td>
<td>97,716</td>
<td>79,648,029</td>
<td>1day 21hr</td>
</tr>
<tr>
<td>No.3</td>
<td>644</td>
<td>134,907</td>
<td>133,208,846</td>
<td>3day 02hr</td>
</tr>
</tbody>
</table>
### Generating Pairings That Allow Breaks

#### Computational Experiments

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Flights</th>
<th>#Arcs</th>
<th>#Valid Pairings</th>
<th>Enum. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>606</td>
<td>123,612</td>
<td>142,777,637</td>
<td>3day 02hr</td>
</tr>
<tr>
<td>No.2</td>
<td>541</td>
<td>97,716</td>
<td>79,648,029</td>
<td>1day 21hr</td>
</tr>
<tr>
<td>No.3</td>
<td>644</td>
<td>134,907</td>
<td>133,208,846</td>
<td>3day 02hr</td>
</tr>
</tbody>
</table>

---

*J. Guo, A. Cohn (UMICH) AAS Presentation Competition Oct. 20, 2019 29 / 37*
### Generating Pairings That Allow Breaks

#### Computational Experiments

<table>
<thead>
<tr>
<th>Datasets</th>
<th>#Flights</th>
<th>#Arcs</th>
<th>#Valid Pairings</th>
<th>Enum. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>606</td>
<td>123,612</td>
<td>142,777,637</td>
<td>3day 02hr</td>
</tr>
<tr>
<td>No.2</td>
<td>541</td>
<td>97,716</td>
<td>79,648,029</td>
<td>1day 21hr</td>
</tr>
<tr>
<td>No.3</td>
<td>644</td>
<td>134,907</td>
<td>133,208,846</td>
<td>3day 02hr</td>
</tr>
</tbody>
</table>

#### Flight Partitioning Heuristic Approach

(Time Limit on DCG is 4hr, on IP is 2hr)

<table>
<thead>
<tr>
<th>Datasets</th>
<th>LP-obj (obj)</th>
<th>#Itr. (Time)</th>
<th>LP Time (Gap)</th>
<th>#Pairings Gen. (Gap)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>551.53 (0.70)</td>
<td>129.5 (4.5)</td>
<td>4.0hr (0.00)</td>
<td>61,600 (4,809)</td>
</tr>
<tr>
<td>No.2</td>
<td>475.28 (4.13)</td>
<td>133.2 (29.5)</td>
<td>1.7hr (0.37)</td>
<td>46,363 (4,393)</td>
</tr>
<tr>
<td>No.3</td>
<td>569.37 (2.71)</td>
<td>132.4 (22.1)</td>
<td>3.6hr (0.65)</td>
<td>58,888 (3,425)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Datasets</th>
<th>IP-obj (obj)</th>
<th>IP Time (Gap)</th>
<th>B&amp;C Gap (%)</th>
<th>Coverage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>494.20 (4.64)</td>
<td>2hr (0)</td>
<td>10.89 (1.07)</td>
<td>81.55 (0.77)</td>
</tr>
<tr>
<td>No.2</td>
<td>426.90 (5.96)</td>
<td>2hr (0)</td>
<td>9.93 (1.21)</td>
<td>78.91 (1.10)</td>
</tr>
<tr>
<td>No.3</td>
<td>510.30 (6.83)</td>
<td>2hr (0)</td>
<td>10.85 (1.12)</td>
<td>79.24 (1.06)</td>
</tr>
</tbody>
</table>
Generating Pairings That Allow Breaks

## Computational Experiments

### Dataset #Flights #Arcs #Valid Pairings Enum. Time

<table>
<thead>
<tr>
<th>No.1</th>
<th>606</th>
<th>123,612</th>
<th>142,777,637</th>
<th>3day 02hr</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.2</td>
<td>541</td>
<td>97,716</td>
<td>79,648,029</td>
<td>1day 21hr</td>
</tr>
<tr>
<td>No.3</td>
<td>644</td>
<td>134,907</td>
<td>133,208,846</td>
<td>3day 02hr</td>
</tr>
</tbody>
</table>

### Flight Partitioning Heuristic Approach

(Time Limit on DCG is 4hr, on IP is 2hr)

#### Dataset LP-obj #Itr. LP Time #Pairings Gen.

<table>
<thead>
<tr>
<th>No.1</th>
<th>551.53 (0.70)</th>
<th>129.5 (4.5)</th>
<th>4.0hr (0.00)</th>
<th>61,600 (4,809)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.2</td>
<td>475.28 (4.13)</td>
<td>133.2 (29.5)</td>
<td>1.7hr (0.37)</td>
<td>46,363 (4,393)</td>
</tr>
<tr>
<td>No.3</td>
<td>569.37 (2.71)</td>
<td>132.4 (22.1)</td>
<td>3.6hr (0.65)</td>
<td>58,888 (3,425)</td>
</tr>
</tbody>
</table>

#### Dataset IP-obj IP Time B&C Gap (%) Coverage (%)

<table>
<thead>
<tr>
<th>No.1</th>
<th>494.20 (4.64)</th>
<th>2hr (0)</th>
<th>10.89 (1.07)</th>
<th>81.55 (0.77)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.2</td>
<td>426.90 (5.96)</td>
<td>2hr (0)</td>
<td>9.93 (1.21)</td>
<td>78.91 (1.10)</td>
</tr>
<tr>
<td>No.3</td>
<td>510.30 (6.83)</td>
<td>2hr (0)</td>
<td>10.85 (1.12)</td>
<td>79.24 (1.06)</td>
</tr>
</tbody>
</table>
## Generating Pairings That Allow Breaks

### Computational Experiments

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Flights</th>
<th>#Arcs</th>
<th>#Valid Pairings</th>
<th>Enum. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>No.1</strong></td>
<td>606</td>
<td>123,612</td>
<td>142,777,637</td>
<td>3day 02hr</td>
</tr>
<tr>
<td><strong>No.2</strong></td>
<td>541</td>
<td>97,716</td>
<td>79,648,029</td>
<td>1day 21hr</td>
</tr>
<tr>
<td><strong>No.3</strong></td>
<td>644</td>
<td>134,907</td>
<td>133,208,846</td>
<td>3day 02hr</td>
</tr>
</tbody>
</table>

**Arc Selection Exact Approach**

- \( H = 250 \)
- Time Limit on IP is 2hr; No Limit on DCG

<table>
<thead>
<tr>
<th>Dataset</th>
<th>LP-obj</th>
<th>#Itr.</th>
<th>LP Time</th>
<th>#Pairings Gen.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>No.1</strong></td>
<td>563.29</td>
<td>41</td>
<td>2hr 09min</td>
<td>79,730</td>
</tr>
<tr>
<td><strong>No.2</strong></td>
<td>492.10</td>
<td>35</td>
<td>1hr 14min</td>
<td>58,448</td>
</tr>
<tr>
<td><strong>No.3</strong></td>
<td>584.42</td>
<td>37</td>
<td>1hr 57min</td>
<td>85,050</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>IP-obj</th>
<th>IP Time</th>
<th>B&amp;C Gap (%)</th>
<th>Coverage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>No.1</strong></td>
<td>521</td>
<td>2hr</td>
<td>7.81</td>
<td>85.97</td>
</tr>
<tr>
<td><strong>No.2</strong></td>
<td>454</td>
<td>2hr</td>
<td>7.89</td>
<td>83.92</td>
</tr>
<tr>
<td><strong>No.3</strong></td>
<td>551</td>
<td>2hr</td>
<td>5.72</td>
<td>85.56</td>
</tr>
</tbody>
</table>
Generating Pairings That Allow Breaks
Computational Experiments

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Flights</th>
<th>#Arcs</th>
<th>#Valid Pairings</th>
<th>Enum. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>606</td>
<td>123,612</td>
<td>142,777,637</td>
<td>3day 02hr</td>
</tr>
<tr>
<td>No.2</td>
<td>541</td>
<td>97,716</td>
<td>79,648,029</td>
<td>1day 21hr</td>
</tr>
<tr>
<td>No.3</td>
<td>644</td>
<td>134,907</td>
<td>133,208,846</td>
<td>3day 02hr</td>
</tr>
</tbody>
</table>

**Arc Selection Exact Approach** \( H = 250 \); Time Limit on IP is 2hr; No Limit on DCG

<table>
<thead>
<tr>
<th>Dataset</th>
<th>LP-obj</th>
<th>#Itr.</th>
<th>LP Time</th>
<th>#Pairings Gen.</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>563.29</td>
<td>41</td>
<td>2hr 09min</td>
<td>79,730</td>
</tr>
<tr>
<td>No.2</td>
<td>492.10</td>
<td>35</td>
<td>1hr 14min</td>
<td>58,448</td>
</tr>
<tr>
<td>No.3</td>
<td>584.42</td>
<td>37</td>
<td>1hr 57min</td>
<td>85,050</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>IP-obj</th>
<th>IP Time</th>
<th>B&amp;C Gap (%)</th>
<th>Coverage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>521</td>
<td>2hr</td>
<td>7.81</td>
<td>85.97</td>
</tr>
<tr>
<td>No.2</td>
<td>454</td>
<td>2hr</td>
<td>7.89</td>
<td>83.92</td>
</tr>
<tr>
<td>No.3</td>
<td>551</td>
<td>2hr</td>
<td>5.72</td>
<td>85.56</td>
</tr>
</tbody>
</table>
Generating Pairings That Allow Breaks
Computational Experiments

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Flights</th>
<th>#Arcs</th>
<th>#Valid Pairings</th>
<th>Enum. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>606</td>
<td>123,612</td>
<td>142,777,637</td>
<td>3day 02hr</td>
</tr>
<tr>
<td>No.2</td>
<td>541</td>
<td>97,716</td>
<td>79,648,029</td>
<td>1day 21hr</td>
</tr>
<tr>
<td>No.3</td>
<td>644</td>
<td>134,907</td>
<td>133,208,846</td>
<td>3day 02hr</td>
</tr>
</tbody>
</table>

**Arc Selection Exact Approach** \( (H = 250; \text{ Time Limit on IP is 2hr}; \text{ No Limit on DCG}) \)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>LP-obj</th>
<th>#Itr.</th>
<th>LP Time</th>
<th>#Pairings Gen.</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>563.29</td>
<td>41</td>
<td>2hr 09min</td>
<td>79,730</td>
</tr>
<tr>
<td>No.2</td>
<td>492.10</td>
<td>35</td>
<td>1hr 14min</td>
<td>58,448</td>
</tr>
<tr>
<td>No.3</td>
<td>584.42</td>
<td>37</td>
<td>1hr 57min</td>
<td>85,050</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>IP-obj</th>
<th>IP Time</th>
<th>B&amp;C Gap (%)</th>
<th>Coverage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1</td>
<td>521</td>
<td>2hr</td>
<td>7.81</td>
<td>85.97</td>
</tr>
<tr>
<td>No.2</td>
<td>454</td>
<td>2hr</td>
<td>7.89</td>
<td>83.92</td>
</tr>
<tr>
<td>No.3</td>
<td>551</td>
<td>2hr</td>
<td>5.72</td>
<td>85.56</td>
</tr>
</tbody>
</table>
Generalization

The arc selection approach would be effective if the density of the underlying network prevents the tractability of the SPPRC model (particularly for scheduling and routing problems). Guaranteed to be exact, if the resource consumption is accumulated in an independently augmenting (possibly with reset) manner.

The flight partitioning approach and the arc selection approach can be integrated together to work (especially when the number of nodes and arcs are both extremely huge). Use the flight partitioning approach to rapidly improve the objective value (the arc selection approach can be applied to solve each instance). Turn to the arc selection approach to help proving optimality.
Generalization

- The **arc selection approach** would be **effective** if the density of the underlying network prevents the tractability of the SPPRC model (particularly for scheduling and routing problems).

- Guaranteed to be exact, if the resource consumption is accumulated in an independently augmenting (possibly with reset) manner.
Generalization

- The **arc selection approach** would be **effective** if the density of the underlying network prevents the tractability of the SPPRC model (particularly for scheduling and routing problems)
  
  - Guaranteed to be exact, if the resource consumption is accumulated in an independently augmenting (possibly with reset) manner.

- The **flight partitioning approach** and the **arc selection approach** can be **integrated together** to work (especially when the number of nodes and arcs are both extremely huge)

  - Use the flight partitioning approach to rapidly improve the objective value (the arc selection approach can be applied to solve each instance)
  
  - Turn to the arc selection approach to help proving optimality
Conclusion and Future Work

Conclusion

- We consider the problem of **generating high-quality crew pairings** to **cover as many scheduled flights as possible** for a cargo airline.

- Two variations are considered, where the advanced case additionally incorporates a “**break**” feature to boost the flight coverage.

- We model the problem as a **set packing problem**, and solve it using a **delayed column generation** framework.

- The pricing problem is formulated as a **SPPRC**, and solved by a **label-setting algorithm** integrated with speed-up improvements.
Conclusion and Future Work

Conclusion

- **A heuristic approach** and an **exact approach** are proposed to address the tractability issue when solving the advanced case.

- Computational experiments on real-world datasets demonstrate the **benefits** of incorporating breaks into the pairing generation, and the **effectiveness** of our proposed approach.

- The proposed heuristic and exact approaches can be used together to potentially solve instances of larger size, and can be generalized and applied to solve a wide range of other scheduling and routing problems.
Conclusion and Future Work

Future Work

- Reduce the B&C gap when solving the IP
  - SDP (i.e. Theta Bag) to provide a tighter upper bound
  - Generate maximal clique cuts

- Apply other heuristics to deal with the integrality constraints (e.g. price-and-dive), or implement the exact B&P framework

- Incorporate deadhead into the pairing generation to further improve flight coverage

- Consider more complicated cost structure of crew pairings
Thank You for Your Attention

Q & A


Reference III


