Railway Crew Pairing with Weekend Considerations: Modeling and Algorithm

Yingwen Chen, Waseda Univ.
Susumu Morito, Waseda Univ.
Jun Imaizumi, Toyo Univ.
Research Background and Objective

• Background:
  – Crew pairing problem studied extensively in railway OR
  – Column generation established as a standard approach

• Difficulties with the traditional approach:
  – Existence of weekday as well as holiday timetables
  – Overnight pairings may not be used unless existence of different timetables is considered

• Objective:
  – Modeling and algorithm development to generate crew pairings which consider holiday timetable
Crew Pairing Problem

Crew pairing problem is a problem of finding a set of pairings which cover all train legs and optimize a certain objective function. Existence of many constraints to be satisfied by each pairing:

- **Leg**: minimum unit of work
- **Pairing**: a sequence of legs crew performs
- **Starting Station (Crew Base)** = **Ending Station (same Crew Base)**
- **Work Hours**
- **Drive Hours**
- **Rest Hours**
- **Sleep Hours of an Overnight Pairing**
- **Number of Train Legs in a Pairing**
- **Earliest Work Start Time, Latest Work Completion Time, Etc.**
Necessity to Consider Holiday Timetable

- Certain train legs may not exist on holiday
Different Types of Overnight Pairings

• Majority of pairings are overnight in practice

• 3 (or 4) different types of overnight pairings
  – Which timetables are used on 1\textsuperscript{st} and 2\textsuperscript{nd} day?
  – Weekday pairing, Holiday pairing, Mixed pairing
Only a few studies refer to existence of different timetables

   - Weekly railway crew pairing generation
   - Difference of timetables depending on day
   - Use of software called TUNRI

   - Airline crew scheduling
   - Difference of weekday and holiday timetables
   - Modifying weekday pairings on holiday
Two-Stage Approach
1. Construct weekday pairings based on weekday timetable
   Use any standard procedure.
2. Construct holiday pairings based on holiday timetable
   Make 1:1 correspondence between a weekday pairing and a holiday pairing.
   - Must satisfy pairing constraints
   - Hope to generate feasible mixed pairings by “concatenation”
Two-Stage Approach

1. Construct weekday pairings based on weekday timetable
   Use any standard procedure.

2. Construct holiday pairings based on holiday timetable
   Make 1:1 correspondence between a weekday pairing and a holiday pairing.
   - Must satisfy pairing constraints
   - Hope to generate feasible mixed pairings by “concatenation”
Two-Stage Approach

1. Construct weekday pairings based on weekday timetable
   Use any standard procedure.

2. Construct holiday pairings based on holiday timetable
   Make 1:1 correspondence between a weekday pairing and a holiday pairing.
   - Must satisfy pairing constraints
   - Hope to generate feasible mixed pairings by “concatenation”
Two-Stage Approach

1. Construct weekday pairings based on weekday timetable
   Use any standard procedure.

2. Construct weekend pairings based on holiday timetable
   Make 1:1 correspondence between a weekday pairing and a holiday pairing.
   - Must satisfy pairing constraints
   - Hope to generate feasible mixed pairings by “concatenation”
Two-Stage Approach

1. Construct weekday pairings based on weekday timetable

   Use any standard procedure.

2. Construct holiday pairings based on holiday timetable

   Make 1:1 correspondence between a weekday pairing and a holiday pairing.
   
   - Satisfy pairing constraints
   - Also generate feasible mixed pairings by “concatenation”

Holiday Pairing Problem

Weekday pairing:

```
A → B → C
```

Midnight

Holiday pairing:

```
A ← C
```

```
C ← A
```

```
C ← B ← A
```

```
A ← B ← C
```
Crew Base and Sleep Station Must Coincide

- Crew base and sleep station of a holiday pairing must coincide with those of the corresponding weekday pairing.

Weekday pairing: A → B → C
Holiday pairing: A → C

Base (Starting and Ending Station)
Sleep Station
Midnight
Conditions to Be Satisfied by Holiday Pairing Problem

(1) The generated holiday pairings satisfy all conditions to be satisfied by pairings.

(2) The concatenated mixed pairings satisfy
   – Crew base constraint
   – Maximum work hours
   – Minimum and maximum sleep hours

(3) The start time of weekend overnight pairing is the same as that of the corresponding holiday pairing.

We also desire the generated a holiday pairing to be similar to the corresponding weekday pairing.
Similarity of a Holiday Pairing and the Corresponding Weekday Pairing

• Similarity measured by
  – Work hours (penalize more work on holiday)
  – Sleep hours (penalize reduced sleep on holiday)

• Similarity is expressed in the objective function of the holiday pairing problem as a “desirability” measure.
Column Generation Algorithm

Start

1. Linealize IP Master (IPM) ⇒ LP Master (LPM)

2. Solve restricted LP Master (RLPM) with a subset of columns of LPM

3. Add columns

   a. Solve column generation subproblem

      i. Check for negative reduced cost

         1. If negative reduced cost, yes
            - Obtain upper bound
            - Dual Price
            - Resource Constrained Shortest Path Problem
            - Lower Bound
            - End
         2. If no, no negative reduced cost ⇒ improve LP solution

There exists one column generation subproblem for each weekday overnight pairing.
Column Generation Subproblem
Resource-Constrained Shortest Path Problem

Network (Node=Leg, Arc=”Connectability” of legs)

Constraints of Mixed Pairing
<table>
<thead>
<tr>
<th>How to satisfy constraints?</th>
<th>How similar pairings are?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Work Hours</td>
<td>Identical start time of pairings <strong>Condition (3)</strong></td>
</tr>
<tr>
<td>Max &amp; Min Sleep Hours</td>
<td>Eliminate certain arcs on the network</td>
</tr>
</tbody>
</table>
Two Variants of Holiday Pairing Problem

• Two variants considered depending on how one-day pairings are generated

• Variant 1:
  – Consider correspondence with weekday one-day pairing
  – (One-day pairing) 1:1 correspondence model

• Variant 2:
  – Do not consider correspondence with weekday one-day pairing; Instead, generate pairings for each crew base
  – (One-day pairing) no correspondence model
Numerical Experiments

Objective

1. Performance evaluation using JR Nara line (Kyoto↔Nara)  
   # of weekday/holiday train legs = 174/162  
2. Comparisons of 2 variants of holiday pairing generation

Method

- PC: Corei7-3770s, 8GB memory, Windows7 64-bit  
- Performed experiments when # of weekday one-day pairings  
  = 1, 2, 3, 4
## Holiday Pairing Problem Solved Quickly

Table 1: Comparison of CPU time (sec.)
 (# of weekday one-day pairings =2)

<table>
<thead>
<tr>
<th></th>
<th>LP Relaxation</th>
<th>Integer Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weekday Pairing</td>
<td>32.04</td>
<td>1440.22</td>
</tr>
<tr>
<td>Holiday Pairing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>One-day 1:1 correspondence</td>
<td>24.04</td>
<td>34.06</td>
</tr>
<tr>
<td>One-day no correspondence</td>
<td>27.55</td>
<td>31.04</td>
</tr>
</tbody>
</table>
“No Correspondence” Model Seems to Yield More Similar Pairings

Table 2: Comparison When # of Weekday One-day Pairings is 2

<table>
<thead>
<tr>
<th></th>
<th>1:1 Corresp.</th>
<th>No Corresp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total # Weekday Pairings</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>Objective Value (① + ②)</td>
<td>40h 41min</td>
<td>33h 43min</td>
</tr>
<tr>
<td>① Diff. of Work Hours</td>
<td>29h 43min</td>
<td>&gt; 23h 52min</td>
</tr>
<tr>
<td>② Diff. of Sleep Hours</td>
<td>10h 58min</td>
<td>&gt; 9h 51min</td>
</tr>
</tbody>
</table>

Table 3: Comparison When # of Weekday One-day Pairings is 3

<table>
<thead>
<tr>
<th></th>
<th>1:1 Corresp.</th>
<th>No Corresp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total # Weekday Pairings</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td>Objective Value (① + ②)</td>
<td>28h 29min</td>
<td>26h 16min</td>
</tr>
<tr>
<td>① Diff. of Work Hours</td>
<td>19h 58min</td>
<td>&gt; 17h 22min</td>
</tr>
<tr>
<td>② Diff. of Sleep Hours</td>
<td>8h 31min</td>
<td>&gt; 8h 54min</td>
</tr>
</tbody>
</table>

No correspondence model allows more flexibility.
Larger # of weekday one-day pairings allows more flexibility.
Conclusions

• Succeeded to generate 4 different sets of crew pairings which take holiday timetable into considerations, based on column generation.

• Appeared that better results could be obtained by allowing more room of flexibility in weekday one-day pairings.

References: