Network Optimization in Transportation Scheduling

Ravindra K. Ahuja
Supply Chain and Logistics Engineering (SCALE) Center
Industrial & Systems Engineering
University of Florida
Gainesville, FL 32611
ahuja@ufl.edu; www.ise.ufl.edu/ahuja
Overview

• Railroad Blocking Problems
  – Motivated by CSX Transportation

• Airline Fleet Scheduling Problems
  – Funded by United Airlines

• Locomotive Scheduling Problems
  – Funded by CSX Transportation
Based on joint research with

Jian Liu
Railroad Blocking Problem

• **Shipments:**
  – Origin-Destination shipments or commodities
  – Size: 50,000 to 100,000 shipments per month
  – Each shipment contains different number of cars
  – Average of 10 cars per shipment

• **Trains:**
  – Thousands of trains per month
  – An O-D shipment is carried by several trains

• **Design the network on which commodities flow.**
Airline Schedule Design Problem

- **Objective**
  - Minimize the travel time
  - Minimize the number of transfers
Design the flight network and route all passengers in it to minimize the weighted sum of travel times and transfers.
Railroad Blocking Problem

Origins

Yards

Destinations

Blocking Arcs
Blocking Problem

• Decision Variables:
  – Design the blocking network
  – Route all shipments over the blocking network

• Constraints:
  – Number of blocking arcs at each node are limited
  – Volume of cars passing through each node is limited

• Objective Function:
  – Minimize the weighted sum of distance traveled by shipment and their intermediate handlings
Size of the Problem

• **Network size:**
  – 1,000 origins
  – 2,000 destinations
  – 300 yards

• **Number of network design variables:**
  – $1,000 \times 300 + 300 \times 300 + 300 \times 2,000 \approx 1$ million

• **Number of flow variables:**
  – 50,000 commodities flowing over 1 million potential arcs
Difficulty of the Problem

• Network design problems are hard nuts.

• Problems with only a few hundred network design variables can be solved to optimality.

• Railroads want a near-optimal and implementable solution within a few hours of computational time.
Prior Research

- Bodin et al. [1980]
- Assad [1983]
- Van Dyke [1986, 1988]
- Newton, Barnhart and Vance [1998]
- Barnhart, Jin and Vance [2000]

- None of the above or any OR approach is used in practice.
Our Approaches

• **Integer Programming Based Methods**
  – Slow and unpredictable

• **Network Optimization Methods**
  – Construction methods
  – Improvement methods
Basic Approach

- Start with a feasible solution of the blocking problem.
- Optimize the blocking solution at only one node (leaving the solution at other nodes unchanged) and reroute shipments.
- Repeat as long as there are improvements.
An Illustration
Basic Approach (contd.)

Out of about 3,000 arcs emanating from a node, select 50 arcs and redirect up to 50,000 shipments to minimize the cost of flow.

We could not solve this problem for one node using CPLEX in one hour.
Basic Approach (contd.)

- We developed a network optimization method to reoptimize the blocking arcs at any node.

- We perform passes over all nodes and reoptimize their blocking arcs one by one.

- We developed a library of reoptimization methods.
### Computational Results

<table>
<thead>
<tr>
<th></th>
<th>Our Solution</th>
<th>CSX Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average car handling</td>
<td>x</td>
<td>1.7x</td>
</tr>
<tr>
<td>Average car miles</td>
<td>x</td>
<td>1.1x</td>
</tr>
</tbody>
</table>

- Considering that about 10 million cars travel annually, the resulting savings are huge. We expect the savings to be over $50 million.
Benefits of Network Based Methods

• Reasonable running times
  – 10-20 minutes
  – Scaleable with the increase in problem size

• Accuracy
  – We believe that our solutions are within 2% - 3% of the optimal solution.

• Flexible
  – Can incorporate a variety of practical constraints.
Future Work

• Working with railroads to identify and incorporate several practical considerations.

• Develop a decision support system for solving the blocking problem.
Additional Applications

• Airline Network Design

• Trucking Network Design

• Package Delivery Network Design
Based on joint research with

Liu Jian
James B. Orlin
Dushyant Sharma

Research supported by

United Airlines
Fleet Assignment Model (FAM)

Assign planes of different types to different flight legs so as to minimize the total cost of assignment.
## Input to Flight Assignment Model

### Inbound Flights at Atlanta:

<table>
<thead>
<tr>
<th>Time</th>
<th>Arrival Time</th>
<th>City</th>
<th>Flight #</th>
<th>Stops</th>
<th>Plane type</th>
</tr>
</thead>
<tbody>
<tr>
<td>6:00 AM</td>
<td>8:05 AM</td>
<td>Boston</td>
<td>709</td>
<td>0</td>
<td>???</td>
</tr>
<tr>
<td>6:30 AM</td>
<td>8:39 AM</td>
<td>JFK</td>
<td>538</td>
<td>0</td>
<td>???</td>
</tr>
<tr>
<td>12:25 PM</td>
<td>4:27 PM</td>
<td>DC</td>
<td>746</td>
<td>0</td>
<td>???</td>
</tr>
<tr>
<td>2:25 PM</td>
<td>6:13 PM</td>
<td>Philly</td>
<td>646</td>
<td>0</td>
<td>???</td>
</tr>
</tbody>
</table>

### Outbound Flights at Atlanta:

<table>
<thead>
<tr>
<th>Time</th>
<th>Departure Time</th>
<th>City</th>
<th>Flight #</th>
<th>Stops</th>
<th>Plane type</th>
</tr>
</thead>
<tbody>
<tr>
<td>6:00 AM</td>
<td>8:05 AM</td>
<td>Houston</td>
<td>657</td>
<td>0</td>
<td>???</td>
</tr>
<tr>
<td>6:30 AM</td>
<td>8:39 AM</td>
<td>Austin</td>
<td>987</td>
<td>0</td>
<td>???</td>
</tr>
<tr>
<td>12:25 PM</td>
<td>4:27 PM</td>
<td>Dallas</td>
<td>564</td>
<td>0</td>
<td>???</td>
</tr>
<tr>
<td>2:25 PM</td>
<td>6:13 PM</td>
<td>Phoenix</td>
<td>367</td>
<td>0</td>
<td>???</td>
</tr>
</tbody>
</table>
## Output of Flight Assignment Model

### Inbound Flights at Atlanta:

<table>
<thead>
<tr>
<th>Time</th>
<th>Flight Time</th>
<th>City</th>
<th>Flight #</th>
<th>Stops</th>
<th>Plane type</th>
</tr>
</thead>
<tbody>
<tr>
<td>6:00 AM</td>
<td>8:05 AM</td>
<td>Boston</td>
<td>709</td>
<td>0</td>
<td>M80</td>
</tr>
<tr>
<td>6:30 AM</td>
<td>8:39 AM</td>
<td>JFK</td>
<td>538</td>
<td>0</td>
<td>757</td>
</tr>
<tr>
<td>12:25 PM</td>
<td>4:27 PM</td>
<td>DC</td>
<td>746</td>
<td>0</td>
<td>M80</td>
</tr>
<tr>
<td>2:25 PM</td>
<td>6:13 PM</td>
<td>Philly</td>
<td>646</td>
<td>0</td>
<td>757</td>
</tr>
</tbody>
</table>

### Outbound Flights at Atlanta:

<table>
<thead>
<tr>
<th>Time</th>
<th>Flight Time</th>
<th>City</th>
<th>Flight #</th>
<th>Stops</th>
<th>Plane type</th>
</tr>
</thead>
<tbody>
<tr>
<td>9:30 AM</td>
<td>11:45 AM</td>
<td>Houston</td>
<td>657</td>
<td>0</td>
<td>757</td>
</tr>
<tr>
<td>9:05 AM</td>
<td>11:00 AM</td>
<td>Austin</td>
<td>987</td>
<td>0</td>
<td>M80</td>
</tr>
<tr>
<td>7:00 PM</td>
<td>9:30 PM</td>
<td>Dallas</td>
<td>564</td>
<td>0</td>
<td>757</td>
</tr>
<tr>
<td>5:30 PM</td>
<td>7:45 PM</td>
<td>Phoenix</td>
<td>367</td>
<td>0</td>
<td>M80</td>
</tr>
</tbody>
</table>
Through Flights

- Combine two flights with the same fleet type passing through a hub into a through flight.
## Additional Through Flights

<table>
<thead>
<tr>
<th>Time</th>
<th>Time</th>
<th>Origin</th>
<th>Destination</th>
<th>Score/$</th>
<th>Plane</th>
</tr>
</thead>
<tbody>
<tr>
<td>6:00 AM</td>
<td>11:00 AM</td>
<td>Boston</td>
<td>Austin</td>
<td>709/987</td>
<td>M80</td>
</tr>
<tr>
<td>6:30 AM</td>
<td>11:45 AM</td>
<td>JFK</td>
<td>Houston</td>
<td>538/657</td>
<td>757</td>
</tr>
<tr>
<td>12:25 PM</td>
<td>7:45 PM</td>
<td>DC</td>
<td>Phoenix</td>
<td>746/367</td>
<td>M80</td>
</tr>
<tr>
<td>2:25 PM</td>
<td>9:30 PM</td>
<td>Philly</td>
<td>Dallas</td>
<td>646/564</td>
<td>757</td>
</tr>
</tbody>
</table>

- Passengers are willing to pay extra for through flights as the same plane flies both the legs.
- Through assignment problem identifies the most profitable matching of inbound and outbound flights.
Current Solution Technique

- When FAM is applied, through revenues are not considered.
- When TAM is applied, fleet assignment cannot be changed.
The Combined Through Fleet Assignment Model (ctFAM)

• Determine fleet assignment and also the through assignment to maximize the total contribution.

• This problem is too large to be solved to optimality or near-optimality by existing integer programming software.
Our Approach for ctFAM

- The improvement algorithm for ctFAM uses very-large scale neighborhood search using A-B swaps.
Single A-B Swaps (Before the swap)
### Single A-B Swaps (After the swap)

<table>
<thead>
<tr>
<th>Orlando</th>
<th>Atlanta</th>
<th>Cincinnati</th>
<th>Boston</th>
<th>New York</th>
<th>Wash. D.C.</th>
<th>Raleigh</th>
</tr>
</thead>
<tbody>
<tr>
<td>8a</td>
<td>8a</td>
<td>8a</td>
<td>8a</td>
<td>8a</td>
<td>8a</td>
<td>8a</td>
</tr>
<tr>
<td>9a</td>
<td>9a</td>
<td>9a</td>
<td>9a</td>
<td>9a</td>
<td>9a</td>
<td>9a</td>
</tr>
<tr>
<td>10a</td>
<td>10a</td>
<td>10a</td>
<td>10a</td>
<td>10a</td>
<td>10a</td>
<td>10a</td>
</tr>
<tr>
<td>11a</td>
<td>11a</td>
<td>11a</td>
<td>11a</td>
<td>11a</td>
<td>11a</td>
<td>11a</td>
</tr>
<tr>
<td>12a</td>
<td>12a</td>
<td>12a</td>
<td>12a</td>
<td>12a</td>
<td>12a</td>
<td>12a</td>
</tr>
<tr>
<td>1p</td>
<td>1p</td>
<td>1p</td>
<td>1p</td>
<td>1p</td>
<td>1p</td>
<td>1p</td>
</tr>
<tr>
<td>2p</td>
<td>2p</td>
<td>2p</td>
<td>2p</td>
<td>2p</td>
<td>2p</td>
<td>2p</td>
</tr>
<tr>
<td>3p</td>
<td>3p</td>
<td>3p</td>
<td>3p</td>
<td>3p</td>
<td>3p</td>
<td>3p</td>
</tr>
<tr>
<td>4p</td>
<td>4p</td>
<td>4p</td>
<td>4p</td>
<td>4p</td>
<td>4p</td>
<td>4p</td>
</tr>
<tr>
<td>5p</td>
<td>5p</td>
<td>5p</td>
<td>5p</td>
<td>5p</td>
<td>5p</td>
<td>5p</td>
</tr>
<tr>
<td>6p</td>
<td>6p</td>
<td>6p</td>
<td>6p</td>
<td>6p</td>
<td>6p</td>
<td>6p</td>
</tr>
</tbody>
</table>

**Type A Plane:**
- From Orlando to Atlanta
- From Cincinnati to Boston
- From Boston to New York
- From New York to Wash. D.C.
- From Wash. D.C. to Raleigh

**Type B Plane:**
- From Atlanta to Orlando
- From Cincinnati to Knoxville
- From Boston to Washington, D.C.
- From New York to Raleigh
Finding Improving A-B Swaps
• Define the cost of each arc as the cost of switching plane types.
• A negative cost cycle gives a profitable A-B swap.
Multi A-B Swaps

Before the swap

After the swap
Identifying Profitable AB-Swaps

• Construct AB-Improvement Graph $G^{AB}(x)$ with respect to the current solution $x$. 

Profitable AB-Swaps with respect to the solution $x$

Negative cost constrained cycles in $G^{AB}(x)$
Neighborhood Search for the ctFAM

• Start with a feasible solution $x$.

• Select two fleet types $A$ and $B$. Construct $AB$-Improvement Graph $G_{AB}(x)$.

• Find negative-cost (constrained) cycle in $G_{AB}(x)$ and update $x$.

• Repeat as long as there are negative cost cycles in $G_{AB}(x)$ for some fleet types $A$ and $B$. 
### Computational Results on ctFAM

<table>
<thead>
<tr>
<th></th>
<th>Increase in Total Contribution (per year)</th>
<th>Running Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local Search</td>
<td>$27.1$ million</td>
<td>5-6 seconds</td>
</tr>
</tbody>
</table>

- United Airline is putting our algorithm and prototype software into production.
ctFAM with Time Windows

- Flight arrival and departure times are also decision variables.

<table>
<thead>
<tr>
<th>Time</th>
<th>Time</th>
<th>Location</th>
<th>Value</th>
<th>Cost</th>
<th>Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>6:00 AM ± 10</td>
<td>8:05 AM ± 10</td>
<td>Boston</td>
<td>709</td>
<td>0</td>
<td>???</td>
</tr>
<tr>
<td>6:30 AM ± 10</td>
<td>8:39 AM ± 10</td>
<td>JFK</td>
<td>538</td>
<td>0</td>
<td>???</td>
</tr>
<tr>
<td>12:25 PM ± 10</td>
<td>4:27 PM ± 10</td>
<td>DC</td>
<td>746</td>
<td>0</td>
<td>???</td>
</tr>
<tr>
<td>2:25 PM ± 10</td>
<td>6:13 PM ± 10</td>
<td>Philly</td>
<td>646</td>
<td>0</td>
<td>???</td>
</tr>
</tbody>
</table>
ctFAM with Time windows (contd.)

- Time windows add greater flexibility to the fleet assignment process.
# Computational Results

<table>
<thead>
<tr>
<th>Local Search</th>
<th>Increase in Total Contribution (per year)</th>
<th>Running Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$45 million</td>
<td>15 minutes</td>
</tr>
</tbody>
</table>
Neighborhood Search in Airline Scheduling

• Can supplement integer programming based approaches.

• We can use neighborhood search to
  – Improve a near-optimal solution
  – To incorporate some additional constraints not satisfied by the solution obtained by integer programming
  – To incorporate non-linearity in the objective function
Locomotive Scheduling Problems

Based on joint research with

Jian Liu, University of Florida, Gainesville
James B. Orlin, MIT, Cambridge
Dushyant Sharma, MIT, Cambridge
Larry Shughart, CSX Transportation, Jacksonville

Funded by

CSX Transportation
Locomotive Schedule Planning Problem

**Given:**
- A set of trains (for a week)
- A set of locomotives

**Determine:**
- Assignment of locomotives to trains

**Satisfying:**
- A variety of constraints

**Minimizing:**
- A sum of cost terms
Some Features

- A train is typically assigned multiple locomotives (called a **consist**).
- Locomotives either actively pull trains or deadhead on them.
- Locomotives can also light travel.
- Trains may not run all days of the week.
Decision Variables

- Active Locomotives
- Deadhead Locomotives
- Light Traveling Locomotives
- Train-Train Connections
Hard Constraints

- Horsepower requirements
- Tonnage requirements
- Fleet size constraints
- Consistency of the assignments
- Consistency of the connections
- Repeatability of the solution
Problem Size

- Number of trains per week: over 3,500
- Number of locomotives: over 2,000
- Number of locomotive types: 5
- Size of the integer programming problem:
  - Number of integer variables: 200,000
  - Number of constraints: 67,000
Two-stage optimization allows us to handle consistency constraints too.
Problem Decomposition

- Determine the three sets of decision variables using a sequential process.
## Computational Results

<table>
<thead>
<tr>
<th>Locomotives Used</th>
<th>LSM (CSX)</th>
<th>ALS (UF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC60</td>
<td>160</td>
<td>160</td>
</tr>
<tr>
<td>AC44</td>
<td>164</td>
<td>154</td>
</tr>
<tr>
<td>C40</td>
<td>621</td>
<td>487</td>
</tr>
<tr>
<td>SD40</td>
<td>498</td>
<td>249</td>
</tr>
<tr>
<td>SD50</td>
<td>171</td>
<td>160</td>
</tr>
<tr>
<td><strong>Total:</strong></td>
<td><strong>1614</strong></td>
<td><strong>1210</strong></td>
</tr>
<tr>
<td><strong>Solution Time:</strong></td>
<td>&gt; 120 min.</td>
<td>20 min.</td>
</tr>
<tr>
<td><strong>Consist Bustings:</strong></td>
<td>85%</td>
<td>49%</td>
</tr>
</tbody>
</table>
### Computational Results (contd.)

<table>
<thead>
<tr>
<th>Locomotives Used</th>
<th>LSM (CSX)</th>
<th>ALS (UF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Time</td>
<td>31%</td>
<td>45%</td>
</tr>
<tr>
<td>Idling Time</td>
<td>49%</td>
<td>46%</td>
</tr>
<tr>
<td>Deadheading</td>
<td>20%</td>
<td>8%</td>
</tr>
<tr>
<td>Light Travel</td>
<td>0%</td>
<td>1%</td>
</tr>
<tr>
<td><strong>Total:</strong></td>
<td><strong>100%</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>
Summary of Computational Results

• Increase in efficiency by about 15%.

• Number of locomotives saved: 400.

• CSX felt that they could save about 50-100 locomotives by the use of this model.
Next Research Phase

• Handling fueling and maintenance constraints.

• Solve the locomotive routing problem.

• We are waiting funding for this step from railroad companies.
Summary

• Substantial savings are possible by the use of optimization methods.

• Solving logistics problems requires insight into network, linear and integer programming techniques.

• Heuristics are critical while solving large-scale logistics optimization problems.
Research Papers

• Most of our papers (or references) are available at:

www.ise.ufl.edu/ahuja/VLSN