Next Generation Decision Support Systems for Railroad Planning

Ravindra K. Ahuja
Professor, University of Florida &
President, Innovative Scheduling
Overview
- Railroad Blocking Optimizer
- Train Scheduling Optimizer
- Locomotive Planning Optimizer
- Overview of Other Systems
- Lessons Learnt
Origin of Innovative Scheduling

- Started research on railroad planning and scheduling problems in 2000.

- The company started its formal operations in 2004 with a single employee and a Phase I grant from NSF’s Small Business Innovations Research (SBIR) Program.

- Received second SBIR Grant in 2005.

- Started commercialization of software immediately.

- Started forming development partnerships with companies to build products.

- The company now has about 20 full-time employees and 8-10 part-time employees.
Our Core Strength

- Ability to solve very complex decision problems efficiently:
  - Blocking problem
  - Train scheduling problem
  - Locomotive planning, simulation problems
  - Crew planning and scheduling problems

- Expertise in a variety of Operations Research techniques:
  - Linear programming
  - Integer programming
  - Network flows and discrete optimization
  - Several heuristic techniques
  - Simulation techniques

- Combine a variety of OR techniques to solve large-scale decision problems very efficiently.
Programming and IT Skills

Programming Skills:
- C++
- Concert Technology, CPLEX
- VB.NET and ASP.NET
- Java
- ESRI GIS programming for maps

Decision Support Systems Building Skills
- Excel-based applications
- Desktop-based applications
- Web-based applications

Most of our solution engines are developed in C++/Java and packaged within web-enabled applications.
Our Railroad Decision Support Systems

- Innovative Railroad Blocking Optimizer (IRBO)
- Innovative Train Scheduling Optimizer (ITSO)
- Innovative Locomotive Planning Optimizer (ILPO)
- Innovative Locomotive Simulation Optimizer (ILSO)
- Innovative Crew Scheduling Optimizer (ICSO)
- Innovative Hump Yard Manager (IHYM)
- Innovative Network Flow Analyzer (INFA)
- Innovative Locomotive Shop Router (ILSR)
- Innovative Yard Simulation Optimizer (IYSO)
Presentation Outline

- Overview

- Railroad Blocking Optimizer
  - Train Scheduling Optimizer
  - Locomotive Planning Optimizer
  - Overview of Other Systems
  - Lessons Learnt
Railroad Planning and Scheduling

- Blocking Problem
- Train Scheduling
- Locomotive Scheduling
- Crew Scheduling
- Yard Operations

Service Design Plan
Consolidation Problem

- Railroad blocking problem is essentially a consolidation problem, which is similar to that encountered in postal service design.

- A railroad block is like a mailbag in the postal service context.
Railroad Blocking Problem

Origins

Yards

Destinations
The Railroad Blocking Model

Network → Railroad Blocking Model → Blocks
Shipments → Railroad Blocking Model → Shipment Block Assignments

**Constraints:**
- Maximum number of blocks that can be build at a node is limited.
- Maximum volume of shipments passing through a node is limited.

**Objective Function:**
- Distances traveled by shipments
- Intermediate handlings of shipments
ABM (Algorithmic Blocking Model) by Carl Van Dyke [1986, 1988]

Keaton [1989, 1992]

Newton, Barnhart and Vance [1998]

Barnhart and Vance [2000]

The railroad blocking problem remained an unsolved problem until recently.
Our Contributions

- Multi-commodity flow network design and routing problem:
  - 3,000 nodes
  - 50,000 commodities
  - Over a million 0-1 network design variables
  - Hundreds of billions of integer flow variables

- We developed a very large-scale neighborhood (VLSN) search algorithm to solve this problem to near-optimality within one-two hours.

- Can also do incremental blocking and handle a variety of practical constraints.
Overview of the VLSN Search Algorithm

- We reoptimize blocks at one node at a time assuming that blocks do not change at other nodes.

- We reoptimize all nodes one-by-one and keep performing passes over the nodes until the solution terminates to a local optimal solution.
### Computational Results: Incremental Blocking

<table>
<thead>
<tr>
<th>% New Blocks</th>
<th>% Savings in Car Miles</th>
<th>% Savings in Intermediate Handlings</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9%</td>
<td>0.5%</td>
<td>7.9%</td>
</tr>
<tr>
<td>1.9%</td>
<td>0.5%</td>
<td>10.5%</td>
</tr>
<tr>
<td>3.8%</td>
<td>0.5%</td>
<td>14.1%</td>
</tr>
<tr>
<td>9.5%</td>
<td>0.6%</td>
<td>19.1%</td>
</tr>
</tbody>
</table>

**Conclusion**: Even small changes in the blocking plan can have significant impact on intermediate handlings.
Railroad Users

Consulting Activities:
- CSX Transportation in One Plan
- Norfolk Southern in TOP II Plan
- BNSF Railway in its current operating plan
- Union Pacific in its Unified Plan

Licensing:
- Norfolk Southern
- BNSF Railway

Potential Future Clients:
- Union Pacific
- Canadian National
- SNCF (France)
- Deutsche Bahn (Germany)
Presentation Outline

- Overview
- Railroad Blocking Optimizer
- Train Scheduling Optimizer
- Locomotive Planning Optimizer
- Overview of Other Systems
- Lessons Learnt
Flow of Blocks on Trains
Train Schedule Design Problem

- Blocks
- Shipments
- Shipment-Block Assignments
- Crew
- Locomotive

Train Scheduling Optimizer

- Trains
- Block-to-Train Assignments
- Trip Plan
- Balanced Crew Assignment
- Balanced Locomotive Assignment
Decision Variables

Decision:
- Train origins, destinations, and routes
- Train days of operation and train times
- Train block-to-train assignment by day of the week
- Trip plans for all cars
- Locomotive assignment
- Crew assignment

Constraints
- Yard capacity constraints
- Line capacity constraints
- Train capacity constraints
- Business rules
We consider these three resources by maintaining three time-space networks.
We construct the weekly time-space train network and flow railcars through this network.
We construct the weekly space-time train network and locomotives cycle through this network.
Crew Scheduling at US Railroads

Each train requires a crew and changes crew at several locations as it travels from its origin to its destination.
We construct the weekly space-time crew network and crews cycle through this network.

We create a separate network for each crew district.
Constraints

❖ Yard Constraints
  ❖ Number of trains originating at any node in each given time window is limited.
  ❖ Number of trains terminating at any node in each given time window is limited.
  ❖ Number of trains passing through each node in each given time window is limited.

❖ Track Constraints
  ❖ Speed of a train on a track depends upon the type of train.
  ❖ Number of trains passing through any corridor in any given time window is limited.
  ❖ Satisfy headway constraints
Constraints (contd.)

- **Train Capacity Constraints**
  - The number of cars on any train is limited
  - The length of any train is limited
  - The weight-carrying capacity of any train is limited
  - No more than specified number of blocks per train
  - Number of stops of a train is limited

- **Locomotive Constraints**
  - Honor locomotive minimum connection times between trains
  - Provide number of locomotive based on train tonnages

- **Crew Constraints**
  - Honor crew minimum connection times between trains
  - Honor crew union rules related to work and rest
Objective Function Terms

- Car miles
- Car days
- Block swaps
- Loco cost
- Crew cost
- Train miles
- Train starts
Our Contribution

Problem size:
- Number of railcars: 125,000
- Number of locomotives: 2,000 – 4,000
- Number of crew districts: 300-400
- Number of crews: 4,000-6,000

We have developed a computer program to solve this problem within 1-2 hours on a laptop.

Uses a variety of operations research techniques:
- Construction heuristics
- Network flows & Linear programming
- Neighborhood search
- Very large-scale neighborhood (VLSN) search
A Two-Stage Decomposition Process

Train Route Optimization

- Train schedule without time
  - Train routes
  - Block-train assignment
  - Locomotive assignment
  - Crew assignment

Train Details Optimization

- Train schedule with time
  - Train routes
  - Block-train assignment
  - Locomotive assignment
  - Crew assignment
Train Route Optimization

- Determine train schedule without train times and day of operation.

**Construction**
- Enumerate all potential train routes
- Determine the goodness of each route
- Select the best route
- Repeat until all blocks are routed

**Improvement**
- Improve routes using neighborhood search
- Improve routes using VLSN search
Train Details Optimization

Train Time Optimization
Decide train arrival and departure times at each stop node on the route

- Optimize one train time by considering all possible options.
- Honor all constraints.
- Repeat for all trains one by one until a local optimal solution is obtained.

Train Operating Days Optimization
Decide operating days of trains

- Optimize one train’s operating days by performing add/drop/exchanges.
- Honor all constraints.
- Repeat for all trains one by one until a local optimal solution is obtained.

Block-to-Train Optimization
Decide block-to-train assignments by day of week

- Optimize one block’s assignment to trains by considering all options.
- Honor all constraints.
- Repeat for all blocks one by one until a local optimal solution is obtained.
Three Time-Space Networks

Railcar Network

Crew Network

Locomotive Network
Two Modes of Train Scheduling

Clean-Slate

- Network, Block and shipment related inputs
  - Optimizer
    - New train schedule, Block-Train Assign., Trip Plans

Incremental

- Network, Block and Shipment Inputs
  - Current Train Schedule
  - Scope of change in train plan
  - Optimizer
    - Revised Train Schedule, Block-Train Assign., Trip Plans
Current Status

- Computational results show 3%-4% improvements in cost.

- Developmental partnership with BNSF. Deployment already started.
Presentation Outline

- Overview
- Railroad Blocking Optimizer
- Train Scheduling Optimizer
- Locomotive Planning Optimizer
- Overview of Other Systems
- Lessons Learnt
Locomotive Planning Optimizer

- Objective Function
- Train Schedules, Tonnages & HP
- Standard Consists
- Locomotive Fleet Description
- Consist Assignment
  - Active power
  - Deadhead power
  - Light travel
- Yard Reports
  - Dwell time
  - Daily supply-demand inventory
  - Train-to-train connections

Constraints
We construct the weekly space-time train network and locomotives cycle through this network.
The Underlying Model

Objective Function
- Locomotive active pulling cost
- Locomotive deadhead cost
- Locomotive idling cost
- Locomotive light travel cost
- Locomotive ownership cost

Standard Constraints
- Each train must get sufficient tonnage and sufficient HP
- Assign only pre-specified consist types to trains
- Honor constraints on min/max locomotives per train
- Allow locomotive imbalances at some nodes
- Honor fleet size requirements
- Weekly repeatable constraints
- Incorporate maintenance constraints
Our Contribution

- We developed a novel methodology to solve this highly constrained problem efficiently using network flows and mixed integer programming.

- We can solve this problem within an hour on a desktop computer.

- Demonstrated a savings of 4% - 5% in the number of locomotives used to run the train plan.
Current Status

- Licensed by CSX Transportation.

- Start sales and marketing to other railroads after completing deployment at CSX Transportation.

- We expect this software to become a standard software for North American railroads.
Presentation Outline

- Overview
- Railroad Blocking Optimizer
- Train Scheduling Optimizer
- Locomotive Planning Optimizer
- Overview of Other Systems
- Lessons Learnt
The business problem:
- All locomotives must go to shops for Q maint. every 92 days
- Sending them earlier results in more Q maint. than necessary
- Sending them late results in dead locomotives
- Have regular and even flow of locomotives to shops consistent with their capacities

What this system will do?
- It will assign locomotives to shops
- It will also assign locomotives to the right train so that they reach the right shop on time

Benefits:
- Even flow of locomotives to shops consistent with their capacities
- Reduction in shop queues and improved shop operations
- Reduction in past due Qs and improved out-of-service rates
Locomotive Demand-Supply Model

The business problem:
- Often terminals get into trouble due to shortage of locomotives
- Locomotive shortage shows up suddenly and unexpectedly
- CSX needs an advance warning system to predict shortages and corrective recommendations and improve train originations

What this system will do?
- Keep track of train movements and locomotive inventories
- Predict excesses and shortages 12-24 hours ahead of time
- In a later phase, provide recommendations for tactical repositionings

Benefits:
- Reduction in terminals getting into trouble
- Improved on-time train originations
Locomotive Simulation Optimizer Engine

- **Event Generators**
  - Train Events
  - Locomotive Events
  - Terminal Events
  - Shop Events

- **Initial State**
  - Trains
  - Locomotives
  - Terminals
  - Shops

- **Locomotive Simulation Model**

- **Decision Engines**
  - Train Arrival/Departure Modules
  - Shop Routing Module
  - Terminal Assignment Module
  - Tactical Repositioning Module

- **Output Database**
Need for crew optimization systems.

Need for crew simulation systems.

We have done significant research and are now seeking development partnerships.
Different types of yards:
  - Hump yard
  - Intermodal yards
  - Flat yards

We are seeking development partnerships to build these systems.
Presentation Outline

- Overview
- Railroad Blocking Optimizer
- Train Scheduling Optimizer
- Locomotive Planning Optimizer
- Overview of Other Systems

Lessons Learnt
## Lessons Learnt from Industry

<table>
<thead>
<tr>
<th><strong>Academia</strong></th>
<th><strong>Industry</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimality</td>
<td>Implementability</td>
</tr>
<tr>
<td>Theoretical Elegance</td>
<td>Return on Investment</td>
</tr>
<tr>
<td>Modeling and Algorithms</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>Mathematical Equations</td>
<td>Visual Information</td>
</tr>
<tr>
<td>Clean-Slate Solution</td>
<td>Incremental Solution</td>
</tr>
<tr>
<td>One-Step Solution</td>
<td>Interactive Solution</td>
</tr>
<tr>
<td>No User Control</td>
<td>Lot of User Control</td>
</tr>
<tr>
<td>Running Time Irrelevant</td>
<td>Real-time Response Time</td>
</tr>
</tbody>
</table>
We become an extension of our customer’s company to build a software product.

We will give our highly discounted rates, comparable to their full-time employees.

Customer receive all design documents and source codes of the software developed under this partnership.

The company retain full rights to use the software for internal purposes and for any yard in its network. It will also retain the rights to extend, modify and enhance the software.
Progressive Development and Deployment

- Decompose a complex decision problem into a sequence of smaller decision problems.
  - Multi-phase development to minimize risk

- Solution of each decision problem creates sufficient return on investment for the client.

- Returns generated from the previous stages fund the future phases.
Others

- Flexibility of relationship
- Role reversal
- Employee ownership
- Patience and persistence
Patience and Persistence