Agent-Based Modeling of Railroad Classification Yards

Ryan Hauser – Undergraduate Student
Natalie Simpson – Faculty Advisor

1. Introduction

A classification yard (pictured on right) is a railroad yard used to separate railroad cars onto one of several tracks for organization to their final destination.

The hump engine (marked 1 at right) is the engine that is used to bring railcars into the yard on open receiving lanes, while the pullback engine (marked 2) is the engine used to pull the railcars back out of the yard onto open departure lanes.

The mentioned engines can select railcars to enter or exit the yard by many criteria (arrival time, block length, scheduled railcar departure time, etc.). Our research focuses on developing robust arrival and departure rules for these engines to utilize in their railcar selection. By implementing the right rules, classification yards can decrease railcar delay, reduce dwell time, and increase overall yard efficiency.

2. The Model

Modeling the Hump Engine

Assuming a hump engine is free to work at minute m, the engine uses assigned inbound rule \( R(k) \) to select train \( t^* \in A_k \) for humping, subject to:

\[
\sum_{k} r_{mk} c_{mk} \leq F, \quad r_{mk} \leq c_{mk}, \quad m = 1, 2, \ldots, M
\]

Where \( r_{mk} \) is the arrival time of train \( t \), \( c_{mk} \) is the inbound inspection time (in minutes), \( A_k \) is the set of all trains in the receiving area at minute \( m \), and \( R \) is the number of receiving lanes.

Modeling the Pullback Engine

Assuming a pullback engine is free to work at minute \( m \), the engine uses assigned outbound rule \( \Phi(k) \) to select train \( k \in K \) for pullback, subject to:

\[
\sum_{k} g_{mk} c_{mk} \leq F, \quad g_{mk} \leq c_{mk}, \quad k = 1, 2, \ldots, K
\]

Where \( g_k \) equals 1 if combination \( k \) requires block type \( b \), and 0 otherwise, \( C_{mk} \) is the number of railcars of block type \( b \) available in the classification lanes at minute \( m \); \( \gamma_{mk} \) is the minimum length of an outbound train, \( \gamma_{mk} \) is the maximum length of an outbound train, \( B \) is the block type, indexed by \( b \), \( D_{mk} \) is the set of all outbound trains in the departure area at minute \( m \), including in-process, and \( F \) is the number of departure lanes.

3. The Experiments

Our experiments involve four 18-day operational scenarios (see figure 1), with that data being provided by the INFORMS Railway Applications Section (RAS). Our modeling is done in C++ in Microsoft Visual Studio 2010 on a personal computer.

We employ a test of twenty-one candidates for \( \Omega(k) \) and nine for \( \Phi(k) \).

Six of the nine candidate \( \Phi(k) \) are rules of our own design, based on the concept of velocity, or choosing an outbound combination according to the rate at which it processes classification lane stock relative to the time invested in the outbound train.

The twenty-one candidates for \( \Omega(k) \) included traditional queue discipline schemes, creative adaptations of mixed model assembly algorithms, and outbound combination scoring schemes.

4. Sample Results (756 simulations, total)

Fig. 2: Average Dwell Time (minutes) for Scenario D

<table>
<thead>
<tr>
<th>Scenario A</th>
<th>Scenario B</th>
<th>Scenario C</th>
<th>Scenario D</th>
</tr>
</thead>
<tbody>
<tr>
<td>482.1</td>
<td>446.3</td>
<td>436.0</td>
<td>493.2</td>
</tr>
<tr>
<td>483.3</td>
<td>433.1</td>
<td>444.0</td>
<td>444.6</td>
</tr>
<tr>
<td>444.6</td>
<td>444.6</td>
<td>446.0</td>
<td>478.8</td>
</tr>
</tbody>
</table>

Fig. 3: Rule Performance vs. Random Diligence

Fig. 4: Yard Population by Rule Pair (Scenario D)

6. Conclusion

Comparing all rule combinations, we find that, on the inbound side, QLT(4) holds a slight advantage across all scenarios. In contrast, the familiar SPT and FCFS are notably poorer choices for selecting trains from the receiving area. Similarly, the highly intuitive outbound rule Q(Grd), or build longest train possible from the available rail car stock, is associated with some of the worst overall yard performance. The fact that these are familiar and intuitive rules, which might conceivably be chosen by rail yard agents in the absence of any other information or direction, raises concern and emphasizes the necessity of supplying these agents with appropriate selection rules. Delving deeper into the outbound side, the productivity-oriented rule \( \Phi(DVL) \) supports the best yard performance across all scenarios. The robustness of our best rules indicates a simpler, more practical alternative to the rolling implementation of mixed integer optimization, with similar (and sometimes superior) yard performance.

6. Honors and References

Semifinalist paper and Honorable Mention in 2013 INFORMS Railways Application Section Problem Solving Competition.

References