Maximizing Trains Platformed, for a given Timetable

Peter Sels$^1,2,3$, Thijs Dewilde$^1$, Dirk Cattrysse$^1$, Pieter Vansteenwegen$^4$

$^1$Katholieke Universiteit Leuven, Centre for Industrial Management/Traffic & Infrastructure, Celestijnenlaan 300a, 3001 Heverlee, Belgium

$^2$Logically Yours BVBA, Plankenbergstraat 112 bus L7, 2100 Antwerp, Belgium e-mail: sels.peter@gmail.com, corresponding author

$^3$Infrabel, Department of Network Access, Frankrijkstraat 91, 1070 Brussels, Belgium

$^4$Ghent University, Department of Industrial Management, Technologiepark 903, 9052 Zwijnaarde, Belgium

June 23, 2011
Table of Contents

1 Business Problem
   - Task

2 Reality To Model

3 Problem Model
   - Sets and Mappings
   - Variables
   - Constraints

4 Results
   - Schedule Graph
   - Solver Response Times

5 Conclusions & Further Work

6 Questions?
Belgian Infrastructure Management Company: Infrabel:
"Maximizing Trains Platformed, for a given Timetable"

Resulting Schedule Implies:
Station Capacity Consumption

Fixed: Constants:
Infrastructure, Timetable, Train Lines, Halting Pattern

To Determine: Variables
Platform, In- & Out-Routing, Corresponding sub-times, (in)feasible trains

Specifics:
One Day of Traffic
Maximizing Trains Platformed, for a given Timetable

Reality To Model

Reality

OUT line
IN line
trhi
trho trto
ntparr
tpdep
tpto
trho trto
=tphi
=trhi

IN routing
OUT routing
platform

IN line
OUT line

space
time

Figure: Train Occupation Time Points and Durations Between Them
Model

Figure: Calculations are done from platform intervals towards routing intervals.
Mixed Integer Linear Programming

- Train Platforming Problem, but with fixed times
- Mixed Integer Linear Programming approach, optimal
- Fictive platform connected to fictive routings, always feasible

Goal Function:

Minimize:

\[ g(op_{o,p}) = \sum_{o \in O_{INI}} C2F_{INI} ch2f_o + C2OR_{INI} ch2or_o + \sum_{o \in O_{SUP}} C2F_{SUP} ch2f_o + C2OR_{SUP} ch2or_o. \]  

(1)

Constraints:

- Infrastructure: Connectivity Platforms-Routings, Routing Conflicts
- Timetable: Fixed Platform Times, Train Run Times, Train Lengths
Infrastructure

- $L$ is the set of lines of both sides of a station, both in and out lines
- $P$ is the set of platforms of a station
- $R$ is the set of routings from lines towards the platforms, and from platforms to lines
- $\forall p \in P : R_p$ is the set of routings that are connected to platform $p$
- $r2p : R \rightarrow P : r \mapsto p$ is the mapping that for each routing $r$, gives the platform $p$ it is connected to
- $dep : R \times R \rightarrow \{0, 1\} : (r_0, r_1) \mapsto dep_{r_0, r_1}$ defines route conflict pairs
Train Activities

- $O$ is the set of occupations to be mapped on platforms
- $M$ is the set of all movements, where several movements can belong to the same occupation
- $M_{IN}$ is the set of IN movements
- $M_{OUT}$ is the set of OUT movements
- $\forall o \in O : M_o$ is the set of movements for an occupation $o$
- $m2o : M \rightarrow O : m \mapsto o$ is the mapping that for each movement $m$, gives the occupation $o$ it is belongs to

Note that occupations of any complexity are supported: stop, pass, split, merge, split & merge, ...
Variables

- \( \forall o \in O \) we define the variables
  - \( o2p_o \) as the platform \( p \in P \) chosen for occupation \( o \)
  - \( \forall p \in P : op_o,p \) as the boolean that is true iff \( o2p_o = p \)

- \( \forall o \in O : \forall m \in M_o \) we define the variables
  - \( m2r_m \) as the routing \( r \in R \) chosen for movement \( m \)
  - \( \forall r \in R : omr_o,m,r \) as the boolean that is true iff \( m2r_m = r \)
Constraints: Allocation

- Occupation to Platform Boolean Integer Variable Relation

\[
\forall o \in O : \left\{ \begin{array}{l}
\sum_{p \in P} op_{o,p} = 1 \\
\sum_{p' \in P} op_{o,p'} \cdot p' = 02p_o.
\end{array} \right. \tag{2}
\]

- Movement to Routing Boolean Integer Variable Relation

\[
\forall o \in O : \forall m \in M_o : \left\{ \begin{array}{l}
\sum_{r \in R} mr_{o,m,r} = 1 \\
\sum_{r' \in R} mr_{o,m,r'} \cdot r' = m2r_m.
\end{array} \right. \tag{3}
\]

- Relation between occupation to platform and movement to routing allocation boolean variables:

\[
\forall o \in O : \forall m \in M_o : mr_{o,m,r} \implies op_{m2o_m,r2p_r} \tag{4}
\]

or/and equivalently:

\[
\forall o \in O : \forall p \in P : op_{o,p} \implies \sum_{r \in R_p} mr_{o,m,r} = 1 \tag{5}
\]
Constraints: Separation

- Separate pair of platform occupation intervals if they are on the same platform resource:

\[
\forall o_0 \in O : \forall o_1 \in O : o_0 \prec o_1 : \\
[otLoLbC_{o_0}, otHiUbC_{o_0}] \cap \\
[otLoLbC_{o_1}, otHiUbC_{o_1}] \neq \phi : \\
\forall p_0 \in P_{o_0} : \forall p_1 \in P_{o_1} : p_1 = p_0 : \\
\frac{op_{o_0, p_0} \land op_{o_1, p_1}}{} \implies o0seo1_{o_0, o_1}.
\]

(6)

- Separate pair of movement routing intervals if they are on the same routing resource:

\[
\forall m_0 \in M : \forall m_1 \in M : m_0 \prec m_1 : \\
[mtLoLbC_{m_0}, mtHiUbC_{m_0}] \cap \\
[mtLoLbC_{m_1}, mtHiUbC_{m_1}] \neq \phi : \\
\forall r_0 \in R_{m_0} : \forall r_1 \in R_{m_1} : dep_{r_0, r_1} : \\
\frac{mr_{o_0, m_0, r_0} \land mr_{o_1, m_1, r_1}}{} \implies m0seo1_{m_0, m_1}
\]

(7)
Maximizing Trains Platformed, for a given Timetable

Results

Schedule Graph

**Schedule Graph**

*Figure*: Occupation Time Interval Graph for Mechelen Station and Peak Original and Optimized Traffic

[Original schedule in dark colors. Optimized schedule in light colors.]*  
Red for initial, blue for supplementary traffic.*
Maximizing Trains Platformed, for a given Timetable

Results

Solver Response Times

Table: Optimization Execution Times I

<table>
<thead>
<tr>
<th>Station</th>
<th>Solver on Machine</th>
<th>#Constraints</th>
<th>#Variables</th>
<th>Time (h,m,s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bergen</td>
<td>Cplex v12.2 on Apple 4C 2.3-3.2 GHz</td>
<td>103362</td>
<td>33473</td>
<td>291.34s</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on Apple 4C 2.3-3.2 GHz</td>
<td>109955</td>
<td>37175</td>
<td>451.35s</td>
</tr>
<tr>
<td></td>
<td>Cplex v11.2 on HP 2C 3.16 GHz</td>
<td>103365</td>
<td>33474</td>
<td>1216s</td>
</tr>
<tr>
<td></td>
<td>Xpress v7.2 on HP 2C 3.16 GHz</td>
<td>109955</td>
<td>37175</td>
<td>OM at 3100s</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on HP 2C 3.16 GHz</td>
<td>109955</td>
<td>37175</td>
<td>4422s</td>
</tr>
<tr>
<td>Brugge</td>
<td>Cplex v12.2 on Apple 4C 2.3-3.2 GHz</td>
<td>34384</td>
<td>10958</td>
<td>3.54s</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on Apple 4C 2.3-3.2 GHz</td>
<td>35627</td>
<td>11717</td>
<td>23.5s</td>
</tr>
<tr>
<td></td>
<td>Cplex v11.2 on HP 2C 3.16 GHz</td>
<td>34384</td>
<td>10958</td>
<td>305s</td>
</tr>
<tr>
<td></td>
<td>Xpress v7.2 on HP 2C 3.16 GHz</td>
<td>35627</td>
<td>11717</td>
<td>11s</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on HP 2C 3.16 GHz</td>
<td>35627</td>
<td>11717</td>
<td>84s</td>
</tr>
<tr>
<td>Dender-leeuw</td>
<td>Cplex v12.2 on Apple 4C 2.3-3.2 GHz</td>
<td>154553</td>
<td>47187</td>
<td>37.57s</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on Apple 4C 2.3-3.2 GHz</td>
<td>159226</td>
<td>50074</td>
<td>26.79s</td>
</tr>
<tr>
<td></td>
<td>Cplex v11.2 on HP 2C 3.16 GHz</td>
<td>154553</td>
<td>47187</td>
<td>621s</td>
</tr>
<tr>
<td></td>
<td>Xpress v7.2 on HP 2C 3.16 GHz</td>
<td>159226</td>
<td>50074</td>
<td>11s</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on HP 2C 3.16 GHz</td>
<td>159226</td>
<td>50074</td>
<td>2651s</td>
</tr>
</tbody>
</table>

Note: #P = number of real platforms. #R = number of real routings. #O = number of occupations, initial and supplementary together. Cplex Matrix dimensions are already (slightly) reduced ones. n.a. = not available (not enough patience limit). OM = Out of Memory.
### Solver Response Times Table II

<table>
<thead>
<tr>
<th>Station</th>
<th>Solver on Machine</th>
<th>#Constraints</th>
<th>#Variables</th>
<th>Time (h,m,s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leuven</td>
<td>Cplex v12.2 on Apple 4C 2.3-3.2 GHz</td>
<td>105709</td>
<td>36242</td>
<td>303s</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on Apple 4C 2.3-3.2 GHz</td>
<td>113119</td>
<td>40676</td>
<td>224.92s</td>
</tr>
<tr>
<td></td>
<td>Cplex v11.2 on HP 2C 3.16 GHz</td>
<td>105709</td>
<td>36242</td>
<td>321s</td>
</tr>
<tr>
<td></td>
<td>Xpress v7.2 on HP 2C 3.16 GHz</td>
<td>113119</td>
<td>40676</td>
<td>3600s (at 3%)</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on HP 2C 3.16 GHz</td>
<td>113119</td>
<td>40676</td>
<td>1569</td>
</tr>
<tr>
<td>Mechelen</td>
<td>Cplex v12.2 on Apple 4C 2.3-3.2 GHz</td>
<td>9959</td>
<td>4756</td>
<td>0.09s</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on Apple 4C 2.3-3.2 GHz</td>
<td>12201</td>
<td>6292</td>
<td>3.1s</td>
</tr>
<tr>
<td></td>
<td>Cplex v11.2 on HP 2C 3.16 GHz</td>
<td>9959</td>
<td>4756</td>
<td>4.3s</td>
</tr>
<tr>
<td></td>
<td>Xpress v7.2 on HP 2C 3.16 GHz</td>
<td>12201</td>
<td>6293</td>
<td>7s</td>
</tr>
<tr>
<td></td>
<td>Gurobi v4.5.1 on HP 2C 3.16 GHz</td>
<td>12201</td>
<td>6293</td>
<td>17s</td>
</tr>
</tbody>
</table>

**Table:** Optimization Execution Times II

#P = number of real platforms. #R = number of real routings. #O = number of occupations, initial and supplementary together. Cplex Matrix dimensions are already (slightly) reduced ones. n.a. = not available (not enough patience limit). OM = Out of Memory.
Maximizing Trains Platformed, for a given Timetable

Results

Solver Response Times

Solver Times Graph

Figure: Intel Core2 Duo, 1.4 GB per process (left) to QuadCore i7, 8GB per process (right) comparison. Serious Improvement across all Solvers.
Conclusions & Further Work

- Conclusions: Business Results
  - \textit{timetable based} capacity (cfr UIC code 406)
  - \textit{fixed time} platforming
  - robust via separation times
  - platforming (& routing) in less than 5 minutes per station
  - both conservative and progressive options
  - improved estimation in practice
  - user surprised about platforming ‘cleverness/inventiveness’
  - business problem solved

  - also fixed time TPP
  - routing choice influences timing and potential conflicts
  - so our conflicts are conditional, 10 to 100 times bigger problems
  - no (conditional) conflict clique heuristics
  - we cover full search space i.o. subset
  - we use real world data i.o. randomized
  - half the solver time
Further Work

- interface
  - availability of data of all stations
- model: consider passenger transfers
  - strive for transfers between neighboring platforms
  - weigh with passenger numbers (flows)
Maximizing Trains Platformed, for a given Timetable

Questions?

sels.peter@gmail.com
References I


